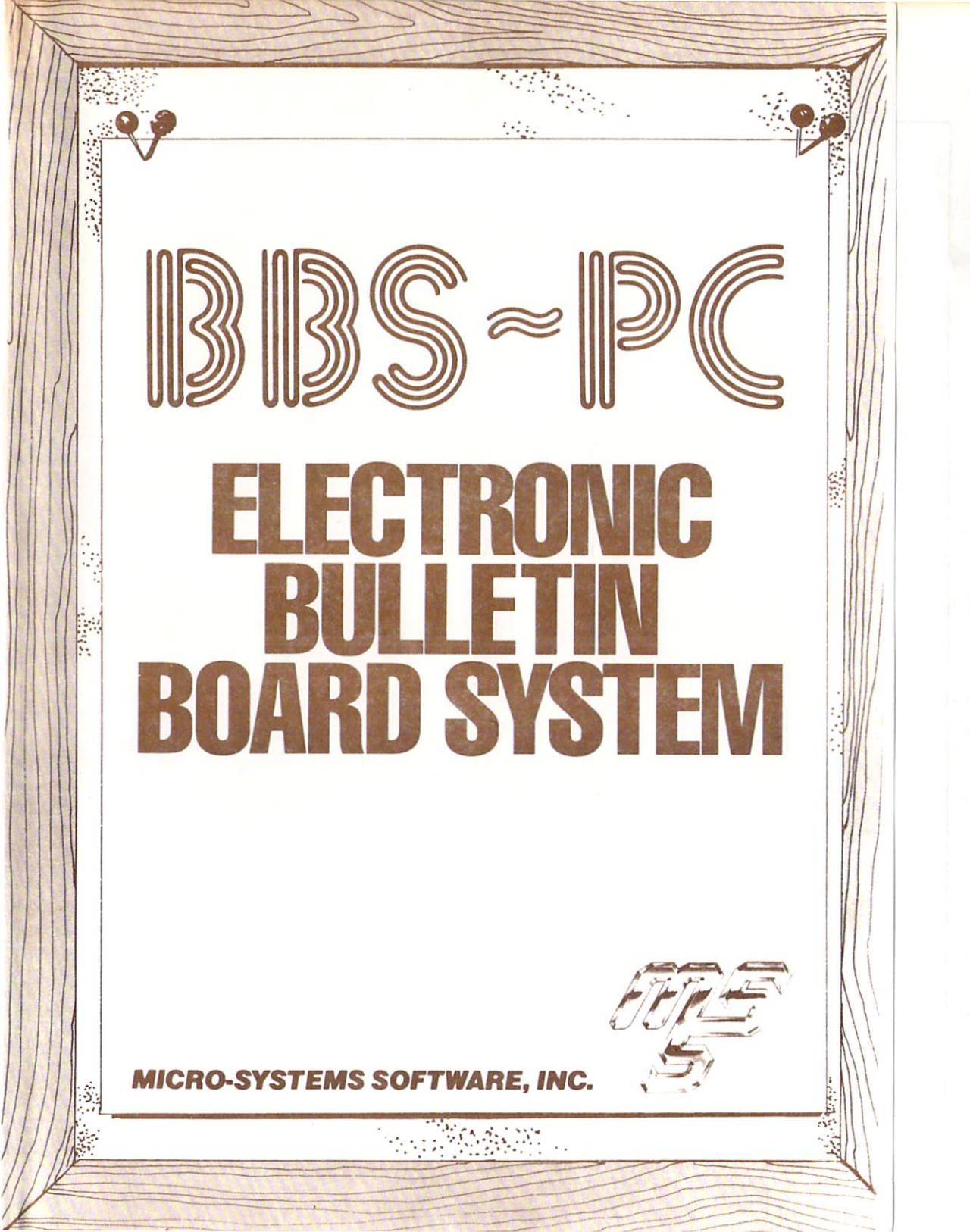


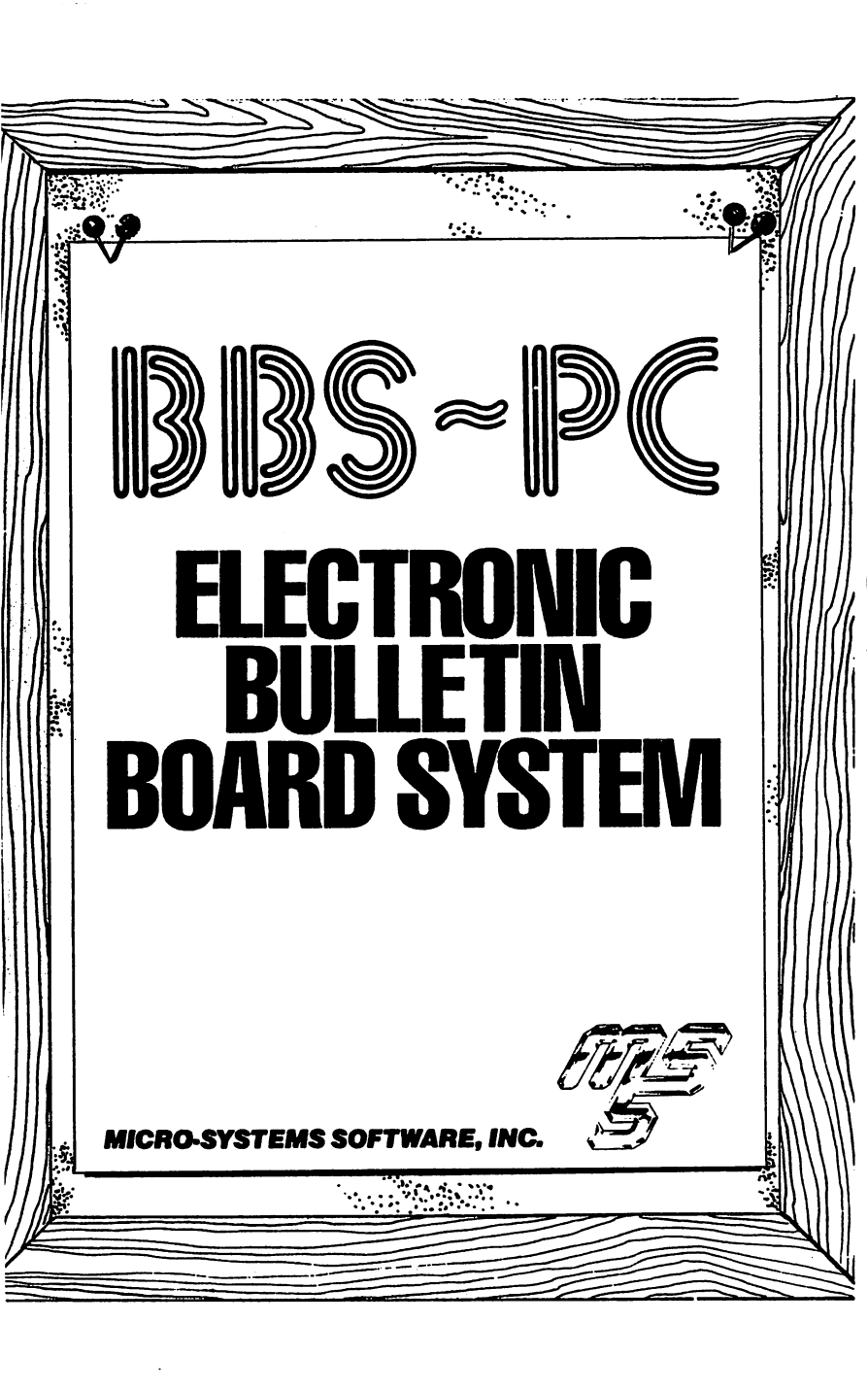


BBS~PC

**ELECTRONIC
BULLETIN
BOARD SYSTEM**



MICRO-SYSTEMS SOFTWARE, INC.



BBS~PC

**ELECTRONIC
BULLETIN
BOARD SYSTEM**



MICRO-SYSTEMS SOFTWARE, INC.

Copyright

BBS-PC! is copyrighted (c) (p) 1985,86 by Micro-Systems Software, Inc. The BBS-PC! Users Manual is copyrighted (c) (p) 1985,86 by Micro-Systems Software, Inc. All rights are reserved. You may not make any copies of the documentation, electronic or otherwise, without the express written permission of the publisher.

Warranty

Micro-Systems Software warrants that this program will perform as documented in this manual, and warrants that the media will be free from defects. This warranty is limited, and Micro-Systems Software specifically disclaims any implied warranties of merchantability or fitness for a particular purpose. Micro-Systems Software's warranty is limited to repair or replacement of the defective media. Micro-Systems Software shall not be liable for any damages, whether consequential or incidental, that arise from the use or misuse of this product. In NO event shall the total amount of Micro-Systems Software's liability exceed the purchase price of the product. No returns accepted after 30 days. All returns may be subject to a re-stocking fee.

Backups and duplicates

Micro-Systems Software grants to the purchaser of this product the right to make duplicates for PERSONAL BACKUP ONLY. The purchaser does not have the right to distribute these duplicates, regardless of whether or not a fee is charged for such distribution. This program is not copy-protected, and may be installed on a hard disk at will. However, installation on multiple CPUs is specifically prohibited. Reasonable corporate site licenses are available.

Technical Support

If you have questions about, or problems with, BBS-PC!, you can write or call:

Micro-Systems Tech Support
4301-18 Oak Circle
Boca Raton, FL 33431

(305) 391-5077

Technicians are available to answer your questions from 9 a.m. until 5 p.m. Monday through Friday, Eastern Standard Time. Less pressing matters can be referred to the BBS.

Bulletin Board Support

Micro-Systems Software maintains a bulletin board online 24 hours a day, 7 days a week. You may call here to ask questions during non-business hours. The bulletin board supports 300, 1200, or 2400 bits per second and any parity/word length setting. The number is:

(305) 737-1590

CompuServe support

Visit us on the CompuServe SIG we SYSOP; the TeleComm Group. Type GO TELECOMM from any prompt and leave a message to SYSOP in section 4 (MSS support). Our PPN is 76703,447, if you want to use EasyPlex.

User's Manual

Table of Contents

Topic.....	pg.
INTRODUCTION.....	1
Conventions Used in this Manual.....	2
Hardware Necessary.....	3
Getting Started.....	3
I. Installation on diskettes.....	4
Moving the data files.....	5
The Default Drive or Directory.....	5
The Message Base.....	5
The User Files.....	5
The Caller Files.....	5
The File Upload/Download Section.....	5
Utility programs.....	6
II. Running the BBSINIT Program.....	7
Configuring your BBS.....	11
The Wait for Caller Menu.....	12
Using the local keyboard.....	13
Accessing the SYSOP Maintenance module.....	15
V: Edit node defaults.....	16
M: Change modem defaults.....	18
N: Display/Edit section names.....	19
W: Edit terminal defaults.....	22
Password protecting the SYSOP account.....	23
That's all, Folks!.....	24
Creating the Text Files.....	25
II. Customizing Your Menus with the BBS-PC! Menu Compiler.....	29
The Menu Statement.....	32
The Title Statement.....	33
The Item Statement.....	34
The Iret Statement.....	36
The Privilege Field.....	36

The Access Field.....	37
Control of Display.....	38
The Endmenu Statement.....	40
The Comment Statement.....	41
Compiling the Menus.....	41
Creating Graphic Menu Sets.....	42
 III. Introduction to BBS-PC! Functions.....	44
 Functions and Menus.....	44
BBS-PC! Functions.....	48
Function 0-User Statistics.....	50
Function 1-Section Names.....	51
Function 2-Change Section Mask.....	53
Function 3-Register/Add to Membership File.....	55
Function 4-Change/Examine User Record.....	57
Function 5-Print Caller Log.....	60
Function 6-Time on System.....	62
Function 7-Chat with SYSOP.....	63
Function 8-Expert Menu Toggle.....	64
Function 9-Leave a Message.....	65
Function 10-Leave Message to SYSOP.....	70
Function 11-Read Messages.....	71
Function 12-Read E-Mail.....	76
Function 13-Scan Messages.....	77
Function 14-Delete a Message.....	79
Function 15-Print File Catalog.....	80
Function 16-Browse Thru Files.....	82
Function 17-Upload a File.....	84
Function 18-Upload a File From Local.....	86
Function 19-Download a File.....	87
Function 20-Read a File.....	88
Function 21-Kill a File.....	89
Function 22-Add a New User.....	90
Function 23-Delete a User.....	92
Function 24-Change a User.....	93
Function 25-Purge Inactive Users.....	95
Function 26-Purge Messages.....	96
Function 27-Print User List.....	97
Function 28-Reset Bulletin Flags.....	98
Function 29-Update User Defaults.....	99
Function 30-Define Section Names.....	100
Function 31-Terminal Defaults.....	102
Function 32-Modem Defaults.....	104
Function 33-User Defaults.....	105
Function 34-System Defaults.....	106
Function 35-Change Node Defaults.....	107
Function 36-List Phone Directory.....	109
Function 37-Change Phone Listing.....	110
Function 38-Dial Connect With Remote.....	111
Function 39-Unlisted Dial/Connect.....	112

Function 40-Upload Direct.....	112
Function 41-Download Direct.....	113
Function 42-Kill a File Direct.....	114
Function 44-Search Catalog Descriptions.....	115
Function 100-Exit System.....	116
Function 101-Display Current Menu.....	117
Function 102-Return Specified Levels.....	118
Function 103-Return Top Level.....	119
Function 104-Call Another Menu.....	120
Function 105-Goto Another Menu.....	121
Function 106-CLS, Type a File.....	122
Function 107-Type File.....	123
Function 108-CLS, Type a File Without "-More (Y/N/NS)-".....	124
Function 109-Type a File Without "-More (Y/N/NS)-".....	125
Function 110-Change Menu Set.....	126

IV. BBS-PC! Support Utility Programs.....128

Introduction.....	128
BBSINIT.COM.....	131
BBSMENU.COM.....	131

The Menu Statement.....	132
The Title Statement.....	133
The Item Statement.....	133
The Iret Statement.....	135
The Privilege Field.....	135
The Access Field.....	136
Control of Display.....	138
The Endmenu Statement.....	140
The Comment Statement.....	140
Compiling the Menus.....	141
Creating Graphic Menu Sets.....	142

BBSINFO.COM.....	144
BBSFIX.COM.....	146
BBSFILE.COM.....	148
CHKFILE.COM.....	150

V. Principles of Operating a BBS.....151

OPEN Systems vs. Restricted Systems.....	153
Defining Your Sections.....	153
The OPEN System.....	155
The RESTRICTED System.....	157
Password Protecting Your SYSOP Account.....	160
Keep Your Users in Mind.....	160

Appendix A - Technical Details and Error Codes.....A-1

File Structures.....	A-1
Error Codes.....	A-9

Appendix B - Special Modem Section..B-1

Introduction.....	B-1
Modem types.....	B-3
Setup string characters.....	B-4

Modems

Hayes.....	B-5
Anchor Automation Signalman Mark XII (aka Volksmodem 12).....	B-5
Radio Shack DC-2212.....	B-6
Radio Shack Model 1000 Internal Modem Board.....	B-7
Ven-Tel Half Card PC Internal Modem.....	B-7
AST Reach Modem.....	B-8
Radio Shack Modem II.....	B-8
Prometheus Promodem 1200.....	B-9
100% Hayes Compatibles.....	B-10
Note to Those Using Pin 6 Sensing.....	B-12

Welcome to BBS-PC!

Introduction

BBS-PC! is the electronic bulletin board system written by Micro-Systems Software for the Commodore Amiga Personal Computer. This manual will provide you with the information needed to install your program and get it running.

An electronic Bulletin Board System (BBS) can serve as a public or private message center, much as a cork bulletin board does in a laundromat, or in a company's lunch room. Users may post public messages and read other public messages. Unlike a cork bulletin board, a BBS can make provisions for private messages which can be read only by the person addressed. Because it operates on a computer connected to a modem, it can be accessed only by users who also have computers, modems and terminal programs. A BBS also offers dozens of other features that cannot be found on cork bulletin boards, such as files of information or programs that can be downloaded from the BBS in much the same manner as one would take a copy of a pamphlet from a rack, or deposit a pamphlet for others to read.

In a business environment, BBS-PC! can serve as an in-house communications center, with each department or branch able to have its own section of the bulletin board. Users may keep abreast of changes in policy, sales goals, inventory and price changes, etc., instantly. BBS-PC! can also be used to transmit daily reports from remote locations to the home office where they may be printed immediately, rather than have to rely on the postal service or voice transmission of reports (with all the attendant errors involved with voice reports). Using appropriate terminal software from the remote locations, such as our own OnLine!, reports may be transmitted with a minimum of operator attendance.

Let's take a quick look at some of BBS-PC!'s advanced features. First time BBS SYSOPs may want to skip over this and proceed to installation, but veteran SYSOPs will find the list useful in comparing BBS-PC! to other bulletin board systems they may already be familiar with.

- * BBS-PC! divides the system into 16 different sections, which can be used for sub-boards or dividing the mail into categories for the main board. The SYSOP (System Operator) controls which users have access to which sections, and even what kind of access users have in those sections (read, write, upload, or download).
- * BBS-PC! offers file transfer using both ASCII and "protocol". Protocols offered include XMODEM (Ward Christensen protocol) XMODEM-STD, XMODEM-CRC, and Hayes Smartcom. The ASCII file transfer mode will send binary files in ASCII-Hex format automatically, and can upload either line by line or in blocks with XON/XOFF pacing.

- * BBS-PC! offers support for high speed modems and automatic detection of a caller's serial parameters. You may tell the program what result codes will be returned by the modem to indicate 300, 1200, or 2400 baud. BBS-PC! will then automatically switch to the correct speed for the incoming call. When a caller connects to the system, BBS-PC! asks him to "Press [RETURN]". When this is done, BBS-PC! will automatically detect the caller's serial parameters (7E1 or 8N1) and adjust accordingly.
- * BBS-PC! allows each caller to configure his terminal's "vital statistics" (such as screen width and length, CLS codes, destructive BS codes, password, expert/novice menu mode) simply by selecting one of a number of terminal types. This creates a pleasant calling environment, especially for callers with smaller screens. Once a caller has selected a terminal type, BBS-PC! will even "word wrap" for him during text entry. Callers may also set and change their own passwords, removing this burden from the SYSOP.
- * For systems requiring close security, BBS-PC! may be set to receive a call, accept user name and password, then disconnect and call back at the number stored in the user's personal record. This helps prevent unauthorized access of the system. The call-back security is set individually for those callers requesting it, or for callers selected by the SYSOP.

In short, BBS-PC! turns your computer into a powerful communications network.

BBS-PC! is not difficult to install or operate; it is not, however, an undertaking for an inexperienced computer user. The more experience a SYSOP has using other BBS', the better he will be able to run his own. He is more familiar with problems that arise and better equipped to understand and remedy them, due mainly to his first-hand experience. You may want to take this under advisement and spend considerable time "learning the ropes" by using other boards before setting up your own.

Conventions used in this Manual

Throughout the manual, the terms "press", "type", and "enter" are used. Each of these carries a slightly different meaning. When you are told to "press ENTER" (for example), you should press the ENTER key. (NOTE: The RETURN key and the ENTER key perform the same exact function.) If you are told to "type L ENTER", you should type those two keystrokes and ONLY those keystrokes. But, if you are told to "enter the current SYSOP password", obviously you will not enter that exact phrase, but rather the correct password. The term "enter" places the responsibility upon you to choose the correct response, usually based upon a list of possible responses.

When a word is surrounded in brackets (e.g. <ENTER>), it refers to pressing a particular key at that point. For example, if you are instructed to "type: BBS <ENTER>", you would type the word "BBS" and press the ENTER key, as opposed to typing literally <-E-N-T-E-R->.

Hardware Necessary

This program operates on an Commodore Amiga Personal Computer with 512K of RAM. It can run with a single floppy disk drive, but this severely limits the capacity of the program in the size of the message base and number of files in the download section. We recommend at least two disk drives, with a hard disk being optimum (as always with programs that have the potential of handling large amounts of data). BBS-PC! uses standard AmigaDOS routines for operating the disk drives, so any hard disk devices compatible with AmigaDOS should be compatible with BBS-PC!. However, while a hard disk system may be optimum for a large download, a very effective BBS can be operated on an Amiga with two floppy disk drives.

The program will operate with almost any command driven modem that will automatically answer the phone and establish carrier. We will devote a portion of this manual to discussing setting up the system for various popular modems, but simply because yours is not mentioned by name does not mean it will not work. BBS-PC! was written to be as flexible as possible in the modem hardware department.

The software defaults assume that you are using a Hayes Smartmodem 1200. If you are using one of these modems (or a compatible unit), you will not need to alter any of the program's modem configurations. Simply set your DIP switches to conform to the list appearing later in this manual and you should be ready to run. If you seem to be having trouble with the modem not behaving correctly, try using a modem type of "4" (as opposed to the Hayes' "0") and verify the SIP switch settings. These are the most common areas of trouble.

Getting Started

To get your system up and running, you will perform the following steps:

1. Prepare the diskettes to hold the data. The program is delivered set up to work on the distribution diskette "as is". However, if it is your desire to use more than one disk drive to increase your bulletin board's capacity, you will need to format these diskettes and have them available in their proper drives prior to proceeding to the next step.
2. If you have made the decision to use additional disk drives, you will need to move the proper data files and run BBSINIT to create a new BBS.P file (BBS.P is the parameter file that tells BBS-PC! where its data files are). For detailed instructions, see "Moving the data files", below.
3. Execute the BBS program. If you are running the program from WorkBench, all you need to do is select the BBS-PC! project icon. Should you wish to start the BBS from CLI, make certain that you are logged into the BBS-PC! subdirectory and have set the stack to 10000, and then you may execute the BBS-PC! program (BBS-PC! is the actual program filename).

4. Once the BBS is running, it still needs to be configured with regards to section names, guest and member defaults, etc. All of these configuration options are contained on the SYSOP Maintenance menu, which you can select by moving the cursor over the "Maintenance" option on the wait for caller menu and pressing RETURN.

This is NOT difficult, and this manual will take you through them one step at a time.

First, before you do anything else, backup the master diskette. BBS-PC! is not copy protected in any manner, so you may use the normal DISKCOPY utility for this. If you are not familiar with CLI, you can use the DiskCopy icon in WorkBench, but this will only perform single drive copies. Users with dual disk drives will find it faster to use the command "DISKCOPY DFO: TO DFI:" (minus the quotes) from CLI. Single drive copies require quite a bit of disk-swapping. And this will always introduce the possibility of an error during the copy process (it is easier than you think to put the wrong disk in at the wrong time).

The program comes pre-configured to use the first disk drive only. If you want to operate a truly effective system, you will need to add at least one, possibly two, additional disk drives. The copy of WorkBench delivered with BBS-PC! is already "stripped", in the interests of conserving space, to provide only commands we considered to be essential (such as Copy and Diskcopy). If you DO only have a single drive, you should copy the BBS-PC! utility programs on to another diskette and delete them from your working master (which should be a backup of the original). This action will save you a great deal of room on the diskette. These programs can then be copied back on to your working master, one at a time, as needed. (Delete them again when you are done.)

Setting Up BBS-PC!

I - Installation on Diskettes

In this section, we will discuss getting your system "on line" (a term used to indicate an operating board). There are some physical things you must do to accomplish this, such as moving the data files around and running the BBSINIT program to tell BBS-PC! where you put them, or moving the utility programs on to another diskette and removing them from the working diskette. There are also certain cosmetic things you'll need to do, such as assigning names to each of the sections on your bulletin board, setting up guest and member defaults, or even editing the menus themselves.

While you do not HAVE to do the cosmetic things (the bulletin board will run without them, using factory defaults), you must get through the physical setup of the board in order for it to work.

It is a good idea to follow this section step by step while you are at your keyboard, so that you may interact with the text. (What we mean by "interact" is that when the manual tells you "type in <command> and you should see <result>", this will make more sense to you if you see it on the display). Again, however, let us remind you that you should really be using a backup copy of the distribution diskette. Perform NO alterations to the original disk unless specifically instructed to by Micro-Systems Software.

Moving the data files

As was mentioned above, BBS-PC! is delivered to you configured to run with one disk drive. This limits the amount of space available, and if you want to run a more effective system, you may employ additional disk drives. You will then have to move some of the data files on to the additional disks. In order to do this, you must first understand how to segregate the data files. A BBS-PC! system can contain up to five divisions. This is not a requirement, and BBS-PC! can run from a single disk drive with all its files in the same subdirectory, but should you decide to move some of the files, you cannot simply move them willy-nilly. You must move an entire "division" of files. The five possible divisions are:

- * The default drive or directory. This should hold the BBS-PC! file (the actual program), all of the required system text files (these files will be covered later), any text files displayed by menu options within the board, the NODExx.DAT file, menus created with the Menu Compiler (BBSMENU), and the CFGINFO.DAT file (the permanent configuration file that contains general system default values).
- * The message base. This should hold all files that start with the three letters "MSG". These files are the indexes and data for the message base. You can backup the message base simply by copying off these files.
- * The user files. This should hold of the files that start with the letters "USER". In these files is kept all of the permanent user records for all of the members of this system (passwords, configurations, etc.).
- * The caller files. Two files keep track of callers to the system and include information on what they did while connected. Both of these start with the letters "CALL".
- * The file upload/download section. This will hold all of the files that start with the letters "UD". These files are the upload/download catalog and the indexes for that catalog. They do not have to be on the same drive as the upload/download files themselves. The location of the actual files is set in another part of the program. This way, a board can have LOTS of files in the catalog, but only as many online as disk space permits. Callers asking for a file not currently online will be told that file is not present and then can make arrangements with the SYSOP.

NOTE: All of the BBS-PC! data files will end with the four characters ".DAT" (minus the quotes, of course). Only these files can be moved. For example, the file UDKEY1.DAT is a BBS-PC! data file and can be moved, along with the rest of the upload/download catalog and indexes. However, the file UDINFO.TXT is a text file displayed by a menu command, and should remain on the same drive as you start BBS-PC! from. Additionally, the file UD.MEN is the default menu for the upload/download section of the board and must also remain on the drive you start BBS-PC! from. When documenting the files that make up the above five divisions, we gave you the common first letters for the latter four divisions. You should also verify, when moving files, that files you move not ONLY have the proper first letters, but end with ".DAT".

Alright, let's take an example. We have a two drive Amiga and want to run BBS-PC!. First thing we do is backup the original diskette (reminders NEVER hurt...). Then, we format a blank diskette in DF1:. When that is ready, we copy the BBS-PC! utility programs on to it. These would be:

```
BBSINIT
BBSFIX
BBSMENU
BBSFILE
BBSINFO
CHKFILE
```

The various jobs these programs do are covered later in manual. For now, we just copy them on to another diskette for storage. Then we delete all of them from the working diskette EXCEPT for BBSINIT. (We will be using that program now and will delete it later.)

Now, we decide that the user files (i.e. passwords and configurations) will remain on DFO:, as will the caller files. But the upload/download files and the message base will go on DF1:. So, the first step is copying all the files that start with "UD" and end with ".DAT" from DFO: to DF1:. Then we copy all the files that start with "MSG" and end with ".DAT" from DFO: to DF1:. Once these data files are safely and successfully copied, we may delete them from DFO:.

This accomplishes the actual physical movement of the disk files. The next task facing us is to run the BBSINIT program and tell BBS-PC! which divisions we moved and where we put them.

II - Running the BBSINIT Program

The BBSINIT program serves four purposes.

1. It creates the BBS.P file. This data file, which must be located on the same drive you start BBS-PC! from, tells the program where to find its five major divisions (discussed earlier).
2. It can create all the data files. These files are on the distribution diskette, and they are empty, so this should not be necessary during your initial setup of the system. But if you decide at some future date that you would like to start over again, it can be useful to have the ability to "wipe the slate clean", so to speak, by deleting the current data files and then having BBSINIT create brand new ones.
3. It tells BBS-PC! where to find the upload/download files themselves. By now, we hope you can see that there is a difference between the download catalog (and the indexes for that catalog), and the actual file to be downloaded. While the actual files can reside on the same drive as the catalog and indexes, they do not have to. (Amiga owners with a third drive may find that they would want to use DF2: as storage for the actual files, while the catalog stays on DF1:. This would allow them to rotate what files are online just by swapping disks in DF2:, since the catalog disk remains online in DF1:.)
4. It creates the NODExx.DAT file (where "xx" is a number between 01 and 99 that describes which "node" we are talking about). BBS-PC! includes the code necessary for multi-user operation. When either a second COM port or some kind of LAN becomes available for the Amiga, it will only require minor program adjustments to support them. When that occurs, these node files will become very important. For right now, you will always be running as node 01. However, even in a single user environment, the node file contains vitally important configuration data. It should not be overlooked.

To run the BBSINIT program, you must exit to CLI. Selecting the "Exit to DOS" function on BBS-PC!'s wait for caller menu will do the trick, if the board is currently running. Type in the program name, "BBSINIT" (minus the quotes, please), and press RETURN.

Amiga will load the program and the BBSINIT program will show:

```
BBSINIT - BBS-PC! Initialization program - 4.0
Copyright (c) 1985, Micro-Systems Software Inc.
```

```
Generate parameter file?
```

This question is asking you if you want to create a BBS.P file. If you have moved any of the data files, you need to do this. We will assume, for the sake of documentation, that you have moved data files and need to give a "Y" answer here. However, if you have not moved any files, please type "N", press RETURN, and proceed to the next question, "Generate main BBS files?".

Drive/path for message files:

This wants to know where the ".DAT" files beginning with "MSG" are. If you have moved these to a drive other than DFO:, type in the proper drive name and press RETURN.

Drive/path for user files :

This prompt is asking you where the ".DAT" files starting with "USER" are (these are the password and configuration files). Again, if you have moved these files, tell BBS-PC! where they are now.

Drive/path for U/D files :

This prompt is seeking the location of the ".DAT" files that begin with "UD" (which are your upload/download catalog and indexes). Once again, if these files are no longer on DFO:, tell BBS-PC! where they are.

Drive/path for log files :

This final location prompt wants to know where the ".DAT" files starting with "CALL" files are. One more time, if you have moved these files from DFO: you'll need to specify their new drive here.

When you tell BBSINIT where those four divisions are (remember that the fifth division is automatically considered the drive and directory from which you start BBS-PC!), you'll see the next prompt.

Generate main BBS files?

This is the next major BBSINIT question. Here the program is asking you if you want to create the data files. Since you already have "empty" data files from the distribution diskette in place, the proper answer here would be "N". In order to be able to respond "Y" here and create the data files, you must first delete the old files. As a safety measure, BBSINIT refuses to overwrite any existing data files.

NOTE: A shorthand "wildmask" command that will delete BBS-PC! data files all at one time without affecting other files on the disk is "Delete ###.DAT" (i.e. you're telling AmigaDOS to delete any files which end with ".DAT", and these should be just the BBS-PC! data files.)

When you answer this question, BBSINIT automatically brings up the next prompt, which is:

Enter up to 16 paths for U/D files
Drive/path #1:

This next series of questions is asking you what drive(s) or directory(s) you will use to hold the actual upload/download files themselves. These are not to be confused with the file catalog indexes and descriptions, which are stored in the location specified at the "U/D files" prompt above. Whenever a caller requests to download a file listed in the catalog, BBS-PC! will look in all 16 of these paths before informing the caller that the file is not on-line. (At which time, the caller should ask the SYSOP make arrangements for the download.)

In addition, all new uploads will be stored in the first of these paths. You may want to use a separate drive or directory for new uploads only.

If you should want to change this list at a later date, you can re-run the BBSINIT program with the "-U" switch (described later on in the manual). As you enter each drive or directory, you will be asked for another.

NOTE: If you are not exactly certain why you would want to segregate the upload/download files, we suggest that you just enter the same drive name you used for the upload/download catalogs and indexes (above). This will serve to minimize confusion, and you can always change them later.

Pressing RETURN alone (with no other input) at one of the prompts will bring you to the final prompt:

Initialize BBS node file?

Responding with a "Y" will display:

Which node (1-99)?

Because this is the primary node, answer with a "1". (The great majority of systems will have only one node, not being part of a network.) Additional nodes in a multi-user system will carry numbers designating their respective locations in the network. Each node can be configured separately with respect to COM ports, modems, and display characteristics.

BBSINIT is documented in further detail later in the manual along with the rest of the utility programs. We just told you as much as you needed to know here. You should read the additional documentation, as well.

Alright, now that we have covered BBSINIT, let's resume with our example bulletin board setup. If you remember, we moved the message base and the upload/download catalog to Df1:. We would run BBSINIT and respond "Y" to the "Generate parameter file?" question. We would answer the prompts like this:

Drive/path for message files: Df1:

Drive/path for user files : DFO:
Drive/path for U/D files : DF1:
Drive/path for log files : DFO:

The next prompt we see asks us if we want to "Generate main BBS files?". Since we are using the data files from the distribution diskette, we answer "N". (Remember that if we wanted to create the files again, we would have to delete the old files.) Amiga accesses the disk drives for a while and then we see the prompt asking us where to look for download files (and where to put new uploads, at the same time). We'll answer like this:

Enter up to 16 paths for U/D files
Drive/path #1: DF1:
Drive/path #2: <RETURN>

This tells BBS-PC! to look for all download files on DF1:, and since the first location is also where new uploads are stored, all new files will be uploaded to DF1:. Had we intended on allowing some of the download files to be on DFO: as well, we could have answered the second prompt with "DFO:" and pressed RETURN at the third prompt.

The last thing we do is answer the "Generate node file?" question with "Y" and the "Which node (1-99)?" question with "1". Once again, Amiga accesses the disk drives for a moment, and then we are returned to DOS. Our job for BBSINIT is done!

You may now delete BBSINIT from the working master, if you like.

III - Configuring your BBS

In the last two sections, we've covered moving the data files on to the correct diskettes and running the BBSINIT program. The combination of these two items completes the physical installation of the software. It will now boot up and run. But our task is not complete. Now we have to concern ourselves with configuring the bulletin board. First, let's see how to start up the program and familiarize ourselves with the "wait for caller" menu.

To start up the program, you have two choices. You can boot up your Amiga using your working master, which will take you into WorkBench. From there, select the BBS-PC! icon. Or, if you wish, you may run the program from CLI. Before running the program from CLI, log into the BBS-PC! subdirectory on your working master. This must be the default directory. (When running the program from WorkBench, the project icon takes care of this task.) In order to run BBS-PC! from CLI, you need two commands:

```
STACK 10000 <RETURN>
BBS-PC! <RETURN>
```

In either case, Amiga will access the disk drives for several seconds and then the screen will clear and BBS-PC! will display its screen. This will appear something like the following example:

Copyright (c) 1985, Micro-Systems Software, Inc.
4301-18 Oak Circle, Boca Raton, FL 33431

```
-----BBS-PC! 4.03-----

Exit program to DOS
Toggle Chat Option
Local BBS Log-in
Upload/Download module
Terminal module
Maintenance module

Chat: OFF
Modem: 0 1

High-msg: 0      Messages: 0
Callers:  0      Users:    1

11:15 AM 10-SEP-86
```


If it does not, there was a problem in the installation. Go back and review carefully. (NOTE: Most of the time, you have moved a file somewhere without correctly running BBSINIT. If you run the BBSFIX utility program, it will report which file cannot be located. You may either then move the file to the proper diskette or run BBSINIT and correct the BBS.P file, whichever needs to be done.)

The Wait for Caller menu

This menu now on your screen is the "wait for caller" menu (sometimes also called the "local" menu). Think of it as sort of the master control panel for your system. Most of your operations will be done from this menu, and it deserves some discussion here. To select an option on this menu, use the arrow keys to position the cursor bar over the option you wish to execute and press the RETURN key. We'll look at each option in turn and see what it does and why you would use it.

"Exit program to DOS" terminates operation of the board and returns to AmigaDOS (or WorkBench, depending on where you were when you started the program). Executing this option also turns off the DTR signal, which should prevent the modem from answering any further calls (at least, this is the way Hayes and truly Hayes compatible modems work).

"Toggle CHAT option" turns the system chat option on and off. "Chat" is a term that means real-time user/SYSOP communication. There is a function on the main BBS menu, as shipped from the factory, to allow callers to signal for a chat. Whenever a caller accesses this function, the board pages the SYSOP by ringing the bell several times. If you care to answer, simply press any key on the keyboard. SYSOP and caller are then in direct communication with each other.

When in Chat, the user and the SYSOP can type freely to each other without having to worry about what they type being taken as a command. If the caller's CRT width and BS codes are set correctly, BBS-PC! will automatically wrap words down to the next line when they will not fit on the current line.

To exit from chat, type a "/" on a blank line and press RETURN.

However, if the SYSOP has no intention of answering chat, the Chat feature can be disabled by pressing RETURN when the cursor is on the "Toggle Chat Option". When the chat command is accessed now, the message "SYSOP not available" will be returned. This serves to let the caller know at once that the SYSOP is not able (or willing) to talk.

Even if chat is turned off, the SYSOP may force a "chat interrupt" that brings the user into the chat mode. Any time the SYSOP presses the F1 key, the user will be brought into the chat mode. The operation that was in progress at the time will be resumed (to best of the system's ability) when the chat mode is exited. Therefore, even if you disable chat, you have the ability to talk to a user if you desire.

Executing this option toggles the chat mode on and off. This will be displayed at the local menu, so you can see what the status is at any time.

The next option down is "Local Log-in". The SYSOP uses this option when he wishes to log into the board to read and answer the mail. This functions exactly as a remote log-in.

Usually, because of the amount of maintenance that needs to be performed on a bulletin board, a SYSOP is his own best customer, so to speak. The SYSOP will be the most frequent caller. Some SYSOPs prefer not to have their name showing up in the caller log over and over.

If a SYSOP wants to avoid being listed in the caller log, there are two options. First, he could alter the caller log privilege range so that someone with a status of 255 is not displayed. Secondly, there is a function on the SYSOP maintenance menu called "Enter BBS through back door". Actually, this is simply a command to take you directly to BBS-O.MEN (which is the first menu after the bulletins). You are never officially logged-in when you do this. (NOTE: This function should only be available on the local SYSOP menu. This is the way the default menus work, but you could alter this without meaning to when you begin editing your menus.)

The final three options simply let you access directly the file transfer, terminal, and SYSOP maintenance modules. You may, if you wish, perform a local log-in and access any of these modules via the commands used for remote access. However, this requires that you answer the log-in prompts. For some applications, direct access is faster and easier. Select the option you want, moving the cursor with the arrow keys, and press RETURN.

Using the local keyboard

There are several special functions available to the SYSOP at the local keyboard that deserve mentioning here. Certain of the ten function keys across the top of Amiga's keyboard have special tasks in BBS-PC!. These are:

<u>Key</u>	<u>Function</u>
F1	Forces an instant "chat interrupt".
F2	Flushes current caller from system instantly.
F3	Grants an additional 15 minutes for this call.
F4	Instant promotion to membership status.
F9	Puts BBS-PC! screen "behind" Intuition screen.
F10	Toggles local printer echo on/off for hardcopy.

The F1 key, chat interrupt, was discussed above. We shall not waste the space doing it again.

Pressing the F2 key while a caller is in the system will IMMEDIATELY disconnect him. It will log them in the caller log as "carrier lost". This can also be useful when the SYSOP is logged in locally. If he hears the BBS phone line ringing, he can press F2, flush himself, and often reset the BBS in time to answer the call.

When a caller reaches two minutes remaining on his access time, BBS-PC! begins counting down the remaining time for him at every input prompt. This is designed to give advance notice to the caller so that they may exit in a controlled manner. If you observe this and wish to extend the caller's access time, for whatever reason, you can press the F3 key. Each time you press the F3 key, the caller will be given another 15 minutes access time. This is ONLY temporary. Any permanent adjustments in a caller's time must be made to their user record. Remember, each press equals 25 minutes. So, 4 presses equals an hour, et cetera. Be careful not to press more times that you want.

If you press the F4 key, the current access defaults that you have defined for members will be assigned to the current caller. This is an instant online membership approval. There is more information regarding guest and member defaults later in this section.

The F9 key will place the BBS-PC! screen "behind" the Intuition screen. You may also accomplish the same task with the normal screen gadgets in the upper right hand corner of the screen, but for those of you with an aversion to mice, the F9 key is provided.

NOTE: When you switch to the Intuition screen, the BBS-PC! project is still active. If you look, you will notice that Intuition is "ghosted" (a term indicating the style of lettering on the screen or window, which in turn indicates whether the program in the displayed window is active or not). You'll need to select the Intuition screen by clicking the mouse inside the screen. If you do not, all keyboard input will still be transferred to the BBS-PC! program, now running invisibly. Also, remember that if it is your intention to operate BBS-PC! completely in the background and run another program at the same time, you will need to either run BBS-PC! from the WorkBench or use the Run command from CLI. If you make BBS-PC! CLI's primary task, you can return to the Intuition screen, but you will be unable to do anything there until you exit BBS-PC!.

The F10 key turns on the hardcopy option. When you press this key, all characters sent to the screen will also be sent to the line printer. To disengage this, press F10 again.

Two other general notes about local operation of the program. First, when a file is entered into the U/D menu, the sender's name is recorded with the file. If you send a file into the menu either remotely or from a local log-in, it will record whatever name you log in with as the sender's name ("SYSOP", your name, or whatever you like). If you send the file into the menu from the U/D maintenance module from the local menu, it will ALWAYS record the name "SYSOP" as the sender.

Second, any time one of these modules is accessed from the local keyboard, the system will use the account defined for SYSOP. Whoever has access to the local keyboard, then also has SYSOP access, at least as far as the maintenance modules are concerned. Use caution with whom you allow access to your computer keyboard.

NOTE: There is one area of operation in which the local and remote modes differ. If you have data in the type-ahead buffer when calling remotely, you can still send control values to pause or abort output and have them recognized. This is because the board looks for these values as it receives each character. However, in the local mode, the board uses the DOS keyboard driver and will not see these control values until all other keystrokes before them are processed. If the system seems to be not responding to a control value when you access it locally, this could be the cause.

Accessing the SYSOP Maintenance module

Although BBS-PC! is now fully operational, there are still really several areas that should be addressed before you can consider your task complete. If you are using a non-Hayes compatible modem, these steps may even be required before you can handle a caller. The physical setup of the bulletin board was just the first part of the battle. Now you will configure certain important program parameters. (After you have run the board for a while with the default menus, you will become ready to begin customizing the program with the menu compiler.)

To accomplish this configuration, you need to access the SYSOP maintenance module from the wait for caller menu. Move the cursor over that option with the arrow keys and press RETURN. You'll see the menu:

```
+-----SYSOP MENU-----+
| A: Add a new user      |
| B: Enter BBS through back door |
| C: Change a user's record |
| D: Delete a user      |
| E: Exit back to main menu |
| L: List current user file |
| M: Change modem defaults |
| N: Display/edit section names |
| O: Purge multiple messages |
| P: Purge inactive users  |
| R: Reset member bulletin flags |
| S: Change user defaults  |
| T: Change system defaults |
| U: Globally edit user access |
| V: Edit node defaults    |
```

! W: Edit terminal definitions !
+-----+

SYSOP Func (? for menu):

NOTE: If your SYSOP account is set up with expert status, the menu will not automatically appear. You will have to, as the prompt suggests, type a "?" and press RETURN to see the menu.

You will not need to use, or know how to use, all these commands at this point. Only certain of them concern the initial configuration of the bulletin board. We will proceed down the line of these options one at a time, looking carefully at each. Any items not listed here may still be important and you can consult the specific function for information, but you will not need it right at this moment.

V: Edit node defaults

BBS-PC! is capable of operating up to 99 nodes on a common database (in a network environment). Although no such environment existed for the Amiga at the time of this writing, the code to support it is already in place. It will require only minor adjustments on our part if and when a network does appear for the Amiga. Because of this, you must configure certain elements about the node you are using now (which will always be node 1 in a single station BBS). The V command is used to set the node defaults. Selecting this option displays the following:

Node defaults:

A: File open mode	: IPL
B: Wake-up on call	: No
C: Local node bell	: On
D: Window frame	: On
E: Active COM lines	:
F: Polled COM lines	:

Enter line to change:

Option A controls when BBS-PC! opens its data files. There are two methods of opening your files. The first is to open them when the program first loads; this is the default, and is indicated by "IPL" (Initial Program Load) at option A. Typing A <RETURN> will display "Open on login (Y/N): " and if you respond with Y, option A will show "Login." The IPL choice opens all files when the program loads, and closes them only when you exit the system properly. The Login choice opens all the data files only when a caller is present and closes them after each call. Opening the files on login takes more time; if you are running a floppy disk system, this could take 20 to 30 seconds (depending on your particular hardware). A caller might think the system has "hung" while waiting for this.

Because the files are open at all times with the IPL option, in the event of a system crash (power failure is the usual culprit), you will probably need to run the BBSFIX program, which is discussed in detail later. The speed of operation with IPL often offsets the inconvenience of having to use the BBSFIX utility to close files before the program will run again, at least on floppy drives. Hard disk users will find that Login should be rapid enough.

Option B, Wake up on call, lets you choose whether you want the BBS-PC! screen to appear "in front of" the Intuition screen automatically when a call is received. After 5 minutes of inactivity while waiting for a call, BBS-PC! will send its own screen behind the Intuition screen automatically. You can optionally configure the system to automatically bring this display back on each time a call is received. Selecting "Y" here does that. Selecting "N" will leave the display off when a call comes in, and you must reactivate it manually. You may bring the BBS-PC! screen back when a caller is connected by pressing any key; it remains on as long as he is on line. (If a call comes in before the display turns itself off again, the display will remain active for the entire call unless you turn it off manually.) You may turn off the display by pressing the F9 key.

Why is this option useful? In some applications, the SYSOP cannot hear the telephone ring or the modem answer. They would never know that a caller is online if the Amiga did no visually alert them. Having the screen come to the front automatically will not select this as the current task, you still need to do that manually. But it WILL serve to alert you to the fact that a caller is present and let you observe their actions.

Option C, Local node bell, configures the system for "silent mode". If you set up BBS-PC! with an "N" here, it will never sound the bell on the local console for any reason. This activates or suppresses ^G, which causes your terminal to beep. Also, an attempt to enter more characters than allowed at a prompt can cause a disturbing racket if this is on.

Option D allows you to choose whether to have the window frame on or off on the bottom (main) window. You might prefer to have it off, to provide to additional columns of display. Certain charts and graphs will be displayed properly only with 80 columns, such as the STATS.TXT file created by BBSINFO.

Option E tells the system which COM ports are active. A "0" indicates COM port 1 and a "1" indicates COM port 2. You may have either or both. Set up this option to reflect how the COM port(s) in your machine are set up. Even if you have two COM ports and only intend to answer calls on one, you should inform this option that you have two. Attempting to access a non-existent COM port later could cause the program to "hang", so BBS-PC! attempts to prevent that. The standard Amiga COM port is 1, which uses a "0" here.

Option F tells the system which COM ports should be monitored to take incoming calls. A "0" indicates COM port 1 and a "1" indicates COM port 2. You may have either or both. Whichever COM port(s) you specify here will be polled for carrier. When the system detects carrier, it proceeds to answer the call. It is possible to have an active COM port that is not checked for an incoming phone call.

These two options need to be set correctly. If you make the mistake of configuring BBS-PC! for a COM port that does not exist, and then exit to DOS, you will not be able to get the program to load again by just typing "BBS" and pressing ENTER. The program will attempt to initialize the COM port that is not there, and will freeze. To get around this problem, there are three switches to use when you call the program from DOS. The program name (BBS-PC!) followed by a space, then one of the following switches:

```
BBS-PC! -C:0
BBS-PC! -C:1
BBS-PC! -C:01
```

This will override polled and active COM lines as defined in options E and F and allow you to reset them.

-C:

This switch tells BBS-PC! that neither COM port is active, but still allows you to boot into the bulletin board. There is also a switch to inform the BBS which node it is using.

-N:x

This switch allows you to select any node from 1 to 99. The default is 1 (which seeks for the file NODE01.DAT).

After you have completed setting the node defaults, press <ENTER> and return to the menu.

M: Change modem defaults

From the menu, type M <ENTER>. The current modem settings will be displayed:

```
Modem #0
A: Default baud: 1200
B: Modem type : 0
C: Setup string: ATSO=1S2=255EQOV1X1|
D: Dial prefix :
E: Dial suffix :
F: 300 result : CONNECT|\
G: 1200 result : CONNECT 1200|\
H: 2400 result : CONNECT 2400|\
Modem #1
I: Default baud: 0
J: Modem type : 0
K: Setup string:
```

L: Dial prefix :
M: Dial suffix :
N: 300 result :
O: 1200 result :
P: 2400 result :

Modem options are broken down into "Modem #0" and "Modem #1". Modem #0 is the modem attached to COM port 1 and Modem #1 is the modem attached to COM port 2. If your serial port is addressed as COM port 2, your modem is still Modem #1, regardless of the fact that it is your only modem. Select the correct modem.

Modem types and their characteristics are discussed in Appendix A. If you are not sure of the correct setup string, dial prefix and suffix for your particular modem, refer to that section of the manual before proceeding.

NOTE: The only reason that you would need to edit the settings BBS-PC! uses by default would be that you have something other than a Hayes 1200 or direct compatible. And the most common change, as mentioned before, is to make the modem type "4". This covers many of the Hayes "not quite" compatible modems. Users with such modems should take the time to carefully read Appendix A at the back of this manual before changing anything here.

N: Display/edit section names

BBS-PC! divides its message base into 16 different sections, numbered 0 through F. The file database has the same 16 sections in it. A user can have access to as many, or as few, of these sections as the SYSOP desires. BBS-PC! allows you to define the default section access for Guests and Members. When someone calls your system for the first time, they will automatically be given the Guest section access. After they register for a password, and you approve them, they will be assigned the Member section access. You may then edit any user's record individually to add or remove a section.

Sections are often used to segregate areas of interest, such as "Dating", "Music", or "Computers". Other times, you can use them to segregate your caller's various computer types, such as "IBM PC", "Apple", or "Commodore". Actually, any time you want to have files and messages grouped together, the sections are the proper way to do this.

There is a direct link between the message base sections and the file database sections. In other words, if you want all the Commodore messages stored in Section 4, the Commodore files should also be in Section 4. The same name will be used. (The file database allows a still further type of segregation using "Directory numbers". That will be discussed in detail later in the manual. For right now, we are concerned only with the sections.)

Callers may also use their "section mask" to voluntarily restrict their access. For example, let's assume that your system is set up with Section 1 for the IBM-PC, Section 2 for TRS-80, and Section 3 for Commodore. John Doe, a regular caller to your system, has access to all three. Normally, when he uses the R command to read mail, he will see messages from all three of these

sections. One day, he decides that he only wants to see messages from Section 3 (he's in a hurry). John can set a section mask of "3" for himself. This time, when John accesses the R command, ONLY messages from Section 3 will be displayed. (NOTE: This is both voluntary and temporary. The caller can restore their own mask to all allowable sections or it will be done for them when they log off. See also: Function 2 - Change section mask.)

Going even deeper into the power of BBS-PC!, the SYSOP can, by editing the menus, have commands on the menu set the section mask automatically. For example, you could have two DIFFERENT commands to read the mail. One command would read Section 1 messages and the other would read Section 3. They would have different keys and descriptions, but call the same Read message function (See: Function 11 - Read messages) just with a differing section mask. This forcing of callers into specific sections creates something known as "sub-boards". This is as contrasted to using the sections as areas of interest in a larger message base. Which style you choose, sub-boards or sections, is up to you. The former requires the use of the menu compiler (BBSMENU) utility program, which may make it a little harder for those still unfamiliar with BBS-PC! to get it working.

Hopefully, we have given you some idea here how sections work. The default menus your BBS-PC! was delivered with will use Section 0 as private EMAIL and Section C for any Comments to the SYSOP. The others are up to you. When you select the N command, you will see the current section structure.

```
0: [_____]
1: [_____]
2: [_____]
3: [_____]
4: [_____]
5: [_____]
6: [_____]
7: [_____]
8: [_____]
9: [_____]
A: [_____]
B: [_____]
C: [_____]
D: [_____]
E: [_____]
F: [_____]

```

Section number:

The prompt is asking for the number of the section you wish to edit. Select one from the list 0-F and enter it. You will then be prompted for the new section name. You may either enter the new name, or press <ENTER> by itself and leave the current name unchanged. The next prompt asks you which of seven possible flags for each section you wish to set. The flags are:

D	Show date in message header.
F	Show message sender in message header.
T	Show addressee in message header.
P	Personal messages allowed in this section.
B	Section bulletin available.
*	Use pseudo name for uploads.
R	Allow addressees to delete mail after they read it.

When you use the menu command to read or scan messages, whether or not the date, to, and from fields are displayed (in addition to the message number, section name, and subject) is controlled by the flags listed above. Therefore, if you want these displayed when callers read or scan messages stored in the section you are naming, enter the proper flags.

The P flag controls whether or not private messages can be stored in that section. Some SYSOPs do not allow private mail in the "general" section. Unless the P flag is set here, a caller attempting to save a message with the "P" option would be told "No private messages." He can then re-direct the message into a different section, save it open, or abort it.

The B flag indicates to BBS-PC! that you intend to have a bulletin for this section. A separate bulletin for each section is allowed, with each user tracked individually with respect to whether or not they have seen it. For every section in which you set a B flag, you must (MUST!) have a bulletin text file in the currently logged drive/directory. (This would be the same directory that you started BBS-PC! from.) The bulletin files take the format BULL-x.TXT (where "x" is the section number; e.g. BULL-1.TXT would be the bulletins for Section 1). It is a common error to set the B flag for a section thinking, "I'll write this bulletin eventually...". If the B flag is present, the BBS will look for this file, and it needs to be there. (See also: Function 28 - Reset bulletin flags.)

NOTE: The actual format of the text files will be discussed in the next section of the manual.

The * flag controls whether or not the file upload/download for that sections uses "pseudo filenames" or not. A pseudo filename is a name created by BBS-PC! to shield the operating system from alien filenames. As an example, AmigaDOS will not allow a slash mark in a filename (since this is used to designate a path). Callers uploading files for the TRS-80 Model III would want to use the slash mark, since that is the TRSDOS file extension separator. Using pseudo filenames engaged will cause BBS-PC! to generate a "FILEnnnn.UDx" name for the file (where "nnnn" is a sequentially incremented 4 digit number and "x" is the node number). To see which pseudo filenames correspond to which index entries, use the CHKFILE utility.

The R flag permits callers to remove (delete) messages in a particular section when (a) they have left the message or (b) the message was left to them. If this flag is not present, only users with SYSOP access in that section will be permitted to delete messages. This is often useful in making a section where replies are shielded from deletion, such as in the case of general technical tips or help with running the board. You can be assured that the messages are still there, since the addressee could not have deleted it inadvertently. (See also: Function 9 - Leave a message.)

Keeping in mind that Guests will have access to Sections 0-3 and Members access additionally to 4-7, a sample section setup might be:

```

0: [DFT---R] ***** EMAIL *****
1: [DFT-B---] Open message base
2: [DFT-----] Items for Sale
3: [-----]
4: [DFTPB-R] Members Only
5: [DFT-----] Commodore Users
6: [DFT-----] IBM-PC Users
7: [DFT-*-] TRS-80 Users
8: [-----]
9: [-----]
A: [-----]
B: [-----]
C: [DFT-----] Comments to SYSOP
D: [-----]
E: [-----]
F: [-----]

```

In the above example, Section 0's name is framed in asterisks to call attention to the fact that there is a private EMAIL awaiting the caller when he logs in and he must use the P command, instead of the normal R command, to read this message. Only Section 4, a message base for general mail between Members, allows private messages. Section 3 is left open for future use when you need a section that Guests will also see. The machine specific sections, which will probably hold the majority of your download files, are in the higher section numbers where Members only get to see them.

Two bulletins are set up, Sections 1 and 4. This requires the files BULL-1.TXT and BULL-4.TXT be present in the currently logged directory.

When you have completed naming the sections you will use and determining how they are to be used by your callers, press <ENTER> to return to the SYSOP command menu.

W: Edit terminal definitions

BBS-PC! maintains a list of ten terminal configuration types that will be presented to the caller when they register for a password. You can save your prospective members a little time this way, as they will not have to edit their user records, at least for the most common terminal types. When you access the W command, you will see:

Name	Page	LF	NULS	CLS codes	BS codes	Protocol
0: Commodore Amiga	75 x 22	Y	0	12 0 0	8 0 0	HVP

1:	80 x 24	Y	0	12 0 0	8 0 0	Text
2:	80 x 24	Y	0	12 0 0	8 0 0	Text
3:	80 x 24	Y	0	12 0 0	8 0 0	Text
4:	80 x 24	Y	0	12 0 0	8 0 0	Text
5:	80 x 24	Y	0	12 0 0	8 0 0	Text
6:	80 x 25	Y	0	12 0 0	8 0 0	Text
7:	80 x 24	Y	0	12 0 0	8 0 0	Text
8:	80 x 24	Y	0	12 0 0	8 0 0	Text
9:	80 x 24	Y	0	12 0 0	8 0 0	Text

Terminal number:

Select the number and enter any changes needed. You will be prompted for:

New name :
 Page width:
 Page len :
 Linefeeds :
 NULS :
 CLS code1 :
 CLS code2 :
 CLS code3 :
 BS code1 :
 BS code2 :
 BS code3 :

BBS-PC! supports more than one code for both clear screen and destructive backspace. It is VERY important that BBS-PC! be able to move the cursor to the left, erasing the character underneath, in order for the word wrap function to operate properly. If only one ASCII code is necessary for the clear screen or the backspace, simply press <ENTER> and move on:

Options:
 T: ASCII Text
 X: Standard XMODEM
 C: XMODEM with CRC
 S: Hayes SmartCom II

Select the default file transfer protocol for this terminal type. They will be able to select a different protocol at download time, but since most people forget, it is useful to see to it that their defaults are correct. After setting up as many terminal types as you wish, press <RETURN> to return to the SYSOP command menu.

Password protecting the SYSOP account

The SYSOP account should be password locked so that outside people cannot call in and access your privileged account. To do this, perform a local login and then access the O command (in the default menus, this is a function 4 - Change/examine user record). The SYSOP record comes into view:

Sysop

```
1: Terminal :  
2: Section  : 0 -  
3: Password  :  
4: Status    : Expert  
5: Protocol  : Text  
6: CLS codes : 12 0 0  
7: BS codes  : 8 0 0  
8: Page size : 76 x 24  
9: Linefeeds : Yes  
0: NULS      : 0
```

Enter line to change:

Select option 3 and enter a password for your account. Then press <RETURN> and exit back to the wait for caller menu.

That's all, Folks!

Your BBS-PC! bulletin board should now be operational and ready to accept its first caller. In the next section of the manual, we discuss creating and editing the various text and help files. Following that, there is a detailed discussion of customizing the menus and a function-by-function documentation of all the BBS-PC! commands.

The final portion of this manual is a piece from a real BBS-PC! SYSOP who gives you some thoughts on the general scheme of being a SYSOP. Sort of a "BBS principles" article. The Modem and Error Code Appendixes bring up the rear. But all of these things can be considered at your leisure, for your BBS-PC! should now be on the air!

If you find that this is not the case, you may want to re-read the text you have just been over, and look for any mistakes. Following that, a call to the Micro-Systems' bulletin board, or perhaps even a call to a technician, might be in order.

Creating the Text Files

An important part of setting up BBS-PC! is to create all the required text files (or modify the samples). BBS-PC! uses ten different text files, which can be identified by the .TXT pattern in the filename. (You will probably want to add others, but the following ten are supplied with the default system.) To see a listing of them, use the command:

DIR ###.TXT

on your BBS-PC! master diskette and directory (which is also named BBS-PC!, incidentally). Among the files shown should be the following names:

- * **CALLBACK.TXT** Displayed when security callback feature is used.
- BBSHELP.TXT** Help with main menu commands.
- BBSINFO.TXT** General information about your board.
- * **LOGIN.TXT** Displayed as a "greeting" as soon as the caller has established carrier.
- * **LOCKOUT.TXT** Displayed whenever a user who has been locked out by the SYSOP attempts to call.
- * **NEWUSER.TXT** Displayed whenever a Guest calls the system, just prior to the standard system bulletins.
- * **BULL-x.TXT** Section bulletin for section x. This file is shown to each member ONCE every time the SYSOP resets the flags. If you have included the bulletin option in setting up your sections, you will need a bulletin for every section that is flagged.
- UDINFO.TXT** Information about what sort of files to upload and download.
- UDHELP.TXT** Help with uploading and downloading.
- * **EDITHELP.TXT** Help with message editor. Displayed by the "H" option of the message leave sub-command prompt.
- * **REGISTER.TXT** Displays information a Guest must enter in order to become a Member.

Each of these contains sample text. The files marked "*" are built-in; the others are defined as part of your menu design. You will undoubtedly want to change the contents of these files; any word processor that will edit and save normal ASCII files may be used.

IMPORTANT NOTE: The following paragraphs that discuss the manner in which the text files should be organized assumes that you have a word processor that edits ASCII files, as opposed to a text editor such as Ed. Micro-Systems is working on such a program, schedule for release in February of 1986. If you do not have access to such a program, Ed may be used, but the files will be hard-formatted, as described below, and the video re-formatting will be defeated. The sample text files sent with the early BBS-PC's, at least, will be hard formatted for Ed already.

All the text files should be formatted in the same manner. What you see is, for the most part, what will appear on the screen. Blank lines in the file are displayed as such on the screen. BBS-PC! will reformat these files during display at the right hand margin to prevent word-splitting. Do not terminate each line with a line feed in these files, as this will defeat the re-formatting. (This is called "hard formatting", when you control the line length with line feeds). Within each paragraph, just let the word processor you are using wrap the lines. (NOTE: Some word processors will save the file with line feeds at the end of the line anyway. If your bulletins appear oddly formatted, this can be the cause.)

If you have entered the text in free format (the term "free format" indicates text with carriage returns or line feeds only at the end of paragraphs and for blank lines), the system will format it smoothly for the caller. The option of hard formatting the file to create short lines is there if you want it, but if users set their page size correctly when they select a terminal type, there will be no need for that technique.

As SYSOP, you can use any of these files for any purpose you desire. It may be of help for you to examine the sample files as we discuss them.

LOGIN.TXT is your greeting screen. This file is displayed to callers as soon as they make the connection to the system. This should identify the board, usually giving a mailing address and voice phone. In addition, this file could contain the SYSOP's name, the hours of operation, and a brief blurb on the system's purpose. This file is formatted to the default page width set by the SYSOP via the SYSOP maintenance module. If the bulk of the callers to your system use "normal" size CRTs, you will want the default page width set to a reasonable size for them. However, you may also know that SOME callers will be using smaller (32 and 40 column) displays. To get around this conflict, set the default page width to 64 or 80 columns, but hard format the LOGIN.TXT file for 32 or 40 columns by placing a carriage return at the end of each line (in the 40th column), forcing the shorter lines.

If the caller is a Guest, the next file he will see is NEWUSER.TXT. In this file should be a "welcome first timer!" message. It might contain information on what the system is all about and who would use it. In addition, it should contain instructions on what the caller should do to become a Member. Tell him at the end that if he wants to read this file again, just enter "B" at the main command menu.

If a user selects "B" from this menu he will be presented with the Bulletin Menu:

```
+-----BULLETIN MENU-----+
! 1: Current Member bulletin   !
! 2: Help with main menu commands !
! 3: General system information !
! 4: Membership information    !
! 5: Information for Guests    !
```

Selection or E)xit:

Selection "1" will display BULL-O.TXT again. Later you will see how this same menu choice can display different bulletins according to the caller's privilege level and access level. A Member will view the appropriate bulletin only once. This is because each Member has a permanent user record which includes a flag indicating he has or has not read the bulletin. If you rewrite bulletins to keep them fresh, you may reset the bulletin flags from the SYSOP Maintenance module.

Selection "2" displays BBSHELP.TXT. This file should hold a brief synopsis of each command on the main menu, but be complete enough that anyone may learn what the commands will do.

Selection "3" displays BBSINFO.TXT, which is usually more philosophical in nature, containing information on the purpose of the board, what sort of material is both available and solicited, and what benefits the Members enjoy. Protocol and general caller information can form a sort of "permanent bulletin" for new users to refer to. It may be a good idea to refer Guests to this file via the NEWUSER.TXT bulletin file. Remember to be brief but clear.

Selection "4" displays a file called SIGNUP.TXT, which should contain all the information someone needs to know to join the system. This is not the same as REGISTER.TXT which is displayed when a caller chooses "A" from the main menu. It will probably be longer and more detailed, setting out all requirements and directing the caller to the "A" command.

Selection "5" will display NEWUSER.TXT again. Notice that Members will not see this on the menu because it is outside their privilege range.

If you have locked out a user by setting his privilege to 0, then he will see the file LOCKOUT.TXT when he attempts to log in, and will then be immediately disconnected. This file should inform him that his access to the system has been revoked, and contain instructions on what to do next (or tell him to get lost, if that's your preference).

The file REGISTER.TXT is displayed as an option from a menu if you want to let Guests register for membership. When a caller selects the option to register, he is presented with this file. The text should spell out the requirements, if any exist, and then prompts him for input of name, address, etc. (See Function 3, Page **** for more details on this.)

The file transfer menu has two commands, "H" and "I", that display the text files UDHELP.TXT and UDINFO.TXT, respectively. The former should have a brief synopsis of each command in the file transfer menu, telling the user what they are for. The latter should be a more general message discussing what sort of files are available for downloading. It can also form a sort of permanent bulletin, informing new users about uploading protocol, and what material may or may not be uploaded, such as copyrighted software.

The file EDITHELP.TXT is displayed from the "H" command on the message leave sub-menu. When you have typed in your message and press ENTER on a blank line to signal the end, you have a menu of leave options. One of them is the "H" option that will display this file. It should contain a simple discussion of what to do next, perhaps detailing a couple of the other sub-menu options in the process.

The file CALLBACK.TXT should be in the default directory or drive if you are going to use the "callback security feature". The text of this file will be displayed just prior to terminating the connection and dialing the caller back. Therefore, this file should contain text explaining the this account has been setup for callback security and that the user should prepare their modem to receive a call. The system will dial them back in 30 seconds.

All of these items were placed into text files so that you could have control over how the system "appears" to your callers. As SYSOP, you have ultimate control over what they contain. You may make them as curt or verbose, as formal or informal, as you please. By combining unique text files with a well-managed section scheme, each SYSOP can create his own atmosphere and theme for the system.

This time, when the local menu (waiting for call) appears on the screen, your system is in business.

Part II

Customizing your Menus with the BBS-PC! Menu Compiler

BBS-PC! is controlled by selecting options from "Menus" which are displayed to the caller. If you think of a Menu as a prompt for the caller to make a selection from a number of available options, you will see that all operations of the BBS are controlled by input from the caller at a Menu. Even a simple prompt asking "Yes or No?" can be displayed only from a Menu. Menus can be as short as "Yes or No?" or as long as you like; they could provide as many choices as you have characters on the keyboard. In practice, most Menus will provide only those choices that are logical at any given point in the program.

In Section III of this manual (BBS-PC! Functions), 56 "Functions" are described, and examples of their use are shown and demonstrated. Each operates as a subroutine within the program; when a caller selects an option from a Menu to exit the BBS, he sees only a single character to select, "E" for example, and enters it at his keyboard. When BBS-PC! receives the "E" it executes "Function 100" which initiates many operations, a few of which are updating his records concerning high message read, time of logoff and the time he spent on the system, how he logged off, how many bytes he uploaded, which bulletins he has read, etc. It also disconnects the phone and performs numerous other tasks such as closing files and resetting itself to wait for another call.

Fortunately, you do not need to be concerned with everything the program does, nor do you need to be an experienced programmer. You need to know only one thing: How do I tell BBS-PC! to perform all those operations? Because we have included all the operations of terminating a call and resetting the system for the next caller in a numbered subroutine ("Function"), you need to know only that "Function 100" is the subroutine to use when the caller wishes to say goodbye. The same is true of any Function.

You may have noticed that the Functions are numbered 0-44, and 100-111. Numbers 45-99 are not "hidden;" they have not been defined, and are reserved for possible future use. They have been classified as General Functions (0-8), general utilities for users; SIG Functions (9-21 and 44), message handling and file transfer; Maintenance Functions (22-35), those operations performed by the SYSOP to keep his system current; Special Functions (36-43), the SYSOP's communication module and remote maintenance operations; and Control Functions (100-111), movements from one menu to another, display of text files, etc.

What Functions you will want to use on each Menu should be determined by what you want to do, or have the user do, when the Menu is displayed. If you think of the Functions as tools, think about grouping them to do a particular task. You would no more want to include phone dialling Functions with message reading Functions than you would want to store electrical tools with lawn maintenance tools.

Years of use and observations of Bulletin Board Systems has shown that this kind of program, more than any other, is customized by the owner to suit his own tastes and desires. It is almost unheard of for a BBS to be operated the way the program was acquired. New features are added, others removed; one will present the caller with an artistic representation of a castle and a greeting that is a full screen in length before asking for the caller's name; another will bluntly ask, "Name?"

Earlier versions of BBS-PC! (up to 3.00) did not provide any opportunity for personal customizing. At best, a few owners manually zapped the program to change some of the messages and prompts displayed by the system. Not only is this tricky business, it still provided no way to customize the appearance or operation of a BBS except for the required text files. Mounting pressure and numerous requests from owners of BBS-PC! resulted in BBS-PC! 4.0 and its companion Menu Compiler. Through months of beta testing, a dozen BBS-PC!'s were started, and today no two of them look or operate alike.

To illustrate the flexibility you have in creating Menus, all three of the following Menus do exactly the same things, despite the different appearance:

CMB-BBS Menu:

- <R>ead Messages
- <E>nter Message
- <U>tilities
- <F>ile Up/Download
- <G>oodbye
- <H>elp
- <I>nformation
- <L>ocal Weather
- <M>ovie Reviews
- <V>alley IBM Club SIG

```
+-- The Hidden Valley Computer Club Message Center -----+
|
| 0 - Read Public Bulletins          5 - HV-IBM Club Members Only  |
| 1 - Post a Public Bulletin        6 - Current Weather Forecast  |
| 2 - File Transfer System          7 - Movie Reviews by the Guru  |
| 3 - Hidden Valley Helps           8 - Hidden Valley Utilities    |
| 4 - Information on Membership      9 - Sign Off for Today         |
|
|----- Contact Ed Jones at 555-1212 for more information ---+
```



R: Read Messages
E: Enter a Message
U: Utilities (User & System)
F: File Transfer Area
G: Goodbye

Other

1. Help with Commands
2. Information about the BBS
3. Local Weather Conditions
4. Movie Reviews by the Guru
5. Hidden Valley IBM Corner

While the first Menu is curt and totally lacking in inspiration, it has the advantage of being able to be displayed correctly on almost any machine. The second Menu, while much more attractive, was designed for an 80-column screen, and is a disaster when displayed on 40-column Apples, Commodores and 64-column Radio Shack machines. The third Menu was carefully planned to fit both the Apples and Commodores, since it is 39-columns wide, retaining the 40th column for the Carriage Return necessary for those machines. However, the underlines display as left arrows on C-64's, unless an alternate set is written utilizing Commodore graphics.

You can customize BBS-PC! to appear and operate totally different from the one next door. This is done through the text files you display to the caller, and what you put on your Menus. Text files are created with any word processor that produces standard ASCII text. Menus are also created with a word processor, or with a text editor. (We recommend our own MSCRIPT and MEDIT.) Text files, such as welcome messages to callers, helps with using the system, and general information, are simply placed on the disk where they can be called and displayed to the caller.

Menus must be compiled before they can be used with BBS-PC!. This is done by using the supplemental program provided called BBSMENU.COM. When you write your Menus, we suggest you save them with an extension of .TXT or .DOC, for example, MENUS.DOC. You can store all your Menus in one source text file, or have a separate file for each. (We'll show advantages of both later.) To compile them, simply enter this at the DOS prompt:

BBSMENU MENUS.DOC

or

bbsmenu menus.doc

The source text in MENUS.DOC will then be compiled, and create one or more Menus with the extension .MEN and write them to the disk. You need a minimum of four Menus, which are:

BBS-0.MEN (Main system Menu)

UD-O.MEN	(Upload/Download Menu)
TM-O.MEN	(Terminal Menu)
SM-O.MEN	(Maintenance Menu)

While these are the minimum required to operate BBS-PC!, you will undoubtedly have many more, such as:

BULL-O.MEN	(Bulletins and Help file Menu)
UTIL-O.MEN	(Utility Menu)
SYSKOM-O.MEN	(Comments to SYSOP Menu)
KAYPRO-O.MEN	(Kaypro SIG Menu)
EXIT-O.MEN	(Exit choices Menu)

Some of these can be very short, such as the Exit Menu, which simply asks, "Do you want to leave us a comment (Y/N)?" as the caller logs off. If you have set aside a section for an IBM-PC User's Group, you could even have a Menu such as PCUG-O.MEN for those users, which is totally different from the Menus seen by other users, and which is as full as your main Menu.

The source text for your Menus contains a list of several things:

A "Menu" Statement
 One or more "Title" Statements
 One or more "Item" Statements
 One or more "Iret" Statements
 An "Endmenu" Statement.

The Menu MUST begin with a Menu Statement and end with an Endmenu Statement. There may be one, none or several of each of the other statements. The following Menu contains one of each except a title:

```
menu = SYSCOM-O.MEN, "Is this private to the SYSOP (Y/N)?", 0
iret = Y, ""\, -, 3:C, 10
item = N, ""\, -, 6, 102, 1
endmenu
```

The Menu Statement

Let's look at the Menu Statement first. It consists of three fields, each of which is separated from the next by a comma. The space following the comma in these examples is optional, but using the space makes the text easier to read.

```
[menu = SYSCOM-O.MEN], "Is this private to the SYSOP (Y/N)?", 0

menu = SYSCOM-O.MEN      This is the "declaration field,"
                        and indicates the beginning of a
                        Menu called "SYSCOM-O.MEN."
```

```
menu = SYSCOM-0.MEN, ["Is this private to the SYSOP (Y/N)?"], 0
```

"Is this private" The "prompt" field, which contains the prompt to be displayed for the Menu. The prompt MUST be enclosed within quotes.

```
menu = SYSCOM-0.MEN, "Is this private to the SYSOP (Y/N)?", [0]
```

0 The "display" option, which states how the Menu will be displayed, if at all.

All Menu Statements must contain those three fields. The third field, the Display Option, defines how the Menu will be displayed:

0 = Never display

1 = Display Menu automatically for Novice.

2 = Display Menu before command prompt for all users.

Display Option 0 allows you to have a prompt with no Menu displayed, as in the example above. Because the only choices are Y and N, there is no need to display anything except the prompt.

Display Option 1 will display a Menu for Novices, but not for Expert Users, who see only the prompt. This is the most frequently-used option for Menus.

Display Option 2 will always display the full Menu regardless of the status of the user. This is generally used on Menus that are seldom used, or in cases where you want to display a brief message prior to getting the user's input.

The Title Statement

A Title is a line that is displayed without defining an item. We could have written the EXIT-0.MEN Menu as follows:

```
menu = SYSCOM-0.MEN, "Is this private to the SYSOP?", 2
title = "Please leave private mail to Members"
title = "in the E-Mail Section."
title = ""
item = N, "N: No.", -, 6, 102, 1
iret = Y, "Y: Yes.", -, 3:C, 10
endmenu
```

When this Menu is displayed, it will look like this:

Please leave private mail to Members
in the E-Mail Section.

Is this private to the SYSOP?

The display option 2 states it will always be displayed, for all users regardless of status. Note that the third Title statement is a null (""); this will display a blank line in the display.

Title Statements may be followed by a privilege range, and this will be discussed below in the "Privilege Field" portion of the Item Statements.

Titles are most commonly used as headers on menus, although they are not confined to that. They may be used to display text anywhere within a menu, or blank lines. Later we'll show you another way of writing a Menu with all display statements created by Titles, and all Item statements hidden from view.

The Item Statement

The Item Statement is the core of the Menu definition. Each Item Statement consists of five or six fields, each of which MUST be separated from the next with a comma; a space following the comma is optional:

```
item = E, "E: Exit System", 10-, 6, 100
item = E, "E: Exit System", -9, 6, 105, EXIT-0.MEN
```

The first field is the Item Definition field, and contains the single letter or other character that will call the Function:

```
[item = E], "E: Exit System", 10-, 6, 100
```

This defines the letter E as the "Key" which will activate the Function in the Function field.

The next field is the Display Description field:

```
item = E, ["E: Exit System"], 10-, 6, 100
```

This is what determines what is displayed on the Menu, and will appear as follows:

```
E: Exit System
```

Following the display field is the privilege level field:

```
item = E, "E: Exit System", [10-], 6, 100
```

item = E, "E: Exit System", [-9], 6, 105, EXIT-O.MEN

In the first example, only a Member (with a privilege level of 10 or greater) will activate Function 100 (Exit system); Guests and New Users (privilege levels below 10) will go to another Menu prior to exiting. This Menu (EXIT-O.MEN) is where he could be asked if he wants to leave a comment.

The fourth field is the Access Field. There are eight levels of access:

- 0 = Requires specified section access
- 1 = Requires specified access with mask
- 2 = Requires RD access with mask
- 3 = Requires WR access with mask
- 4 = Requires DN access with mask
- 5 = Requires UP access with mask
- 6 = No section access required
- 7 = Requires SYSOP password, if set.

Neither of the Exit items above requires particular section access, and is available to anyone within the proper privilege levels.

Before we show examples of other Access levels, let's continue with the next field, the Function Field:

item = E, "E: Exit System", 10-, 6, [100]
item = E, "E: Exit System", -9, 6, [105], EXIT-O.MEN

In the first example, choosing the option E will activate Function 100 (Exit system). The second example activates Function 105 (Goto another menu), and requires a final, or sixth field, called the Argument Field:

item = E, "E: Exit System", -9, 6, 105, [EXIT-O.MEN]

The Argument Field contains the name of a program, data or text file or may contain a numeric constant. Only the following Functions admit an Argument:

- 102 - Return specified levels
- 104 - Call another menu
- 105 - Goto another menu
- 106 - CLS, type a file
- 107 - Type a file
- 108 - CLS, type a file without "- more (Y/N/NS)"
- 109 - Type a file without "- more (Y/N/NS)"
- 110 - Change menu set
- 111 - Execute external program

These Functions, and only these Functions, will be followed by an Argument. The Argument, whether a Menu or Text File, is never enclosed within quotes.

The Iret Statement

An Iret Statement consists of the same fields as an Item Statement. In the SYSCOM-0.MEN above, it appears this way:

```
iret = Y, "\"\, -, 3:C, 10
```

The only difference between an Item and an Iret, is that the Iret Statement means, "Perform the Function indicated, then RETURN to the Menu from which this Menu was called." Because only the prompt is seen offering a choice of Yes or No in this example, there is no need to display anything else. If this were to display anything, the Display Description field would be filled in within the quotes.

The Privilege Field

The third field in an Item Statement or Title Statement is the Privilege Field. This determines not only who can select the option, but also who can even see it displayed. Take the following for example:

```
item = E, "E: Exit System", 10-, 6, 100
title = "-----", -, -9
title = "Guests are asked to leave a comment", -, -9
title = "when you exit. We'd like to know how", -, -9
title = "you like this system.", -, -9
title = "-----", -, -9
item = E, " E: Exit System", -9, 6, 105, EXIT-0.MEN
```

Of the eight lines, a Member (Privilege 10 or greater) will see only the first line displayed:

```
E: Exit System
```

New Users and Guests (anyone whose Privilege is less than 10) will see:

```
+-----+
|Guests are asked to leave a comment|
|when you exit. We'd like to know how|
|you like this system.              |
+-----+
E: Exit System
```

In this manner, Guests and Members see different parts of the Menu, and with multiple Item Statements as shown here, selecting the option E will bring different Functions into play. Your only limitation is that you may have a total of no more than 50 Item, Iret and Title Statements in any one Menu. Seldom will this many be necessary, for your Main Menu could direct Members and Guests to different Menus for the duration of their call on the board.

The Access Field

The fourth field in an Item or Iret Statement is the Access Field, and determines what type of access is necessary to activate the Function. The access levels are:

- 0 = Requires specified section access
- 1 = Requires specified access with mask
- 2 = Requires RD access with mask
- 3 = Requires WR access with mask
- 4 = Requires DN access with mask
- 5 = Requires UP access with mask
- 6 = No section access required
- 7 = Requires SYSOP password, if set.

The first six Access Levels require setting the specific mask as well as the Section Number. The Section Number is appended to the Access Level Number with a colon:

```
item = R, "R: Read Messages", -, [2:-BC], 11
item = I, "I: IBM SIG", 25-, [2:B], 104, PCUG-O.MEN
item = K, "K: Kaypro SIG", 25-, [2:C], 104, KUG-O.MEN
```

Note that in the first example, Sections B and C are removed (with the "-"), allowing all other sections to be read (B and C are "masked out"). In the next two examples, which are seen only by users with a privilege level of 25 or higher), Sections B and C are "masked in" and the other sections are masked out. If a user does not have Read Access to these sections, he will not even see the option displayed.

In the Exit examples shown, no specific section access is required, as shown by the "6" in the field:

```
item = E, "E: Exit System", 10-, [6], 100
item = E, "E: Exit System", -9, [6], 105, EXIT-O.MEN
```

Often you will want to limit what a user may do by restricting the access level. For example, if you set section B aside for a SIG such as the PC Users Group for members only, you could prevent other users from seeing any messages in that section in the following manner:

```
item = R, "R: Read Messages", -, [2:-B], 11
```

The access level of 2 indicates the user must have Read Access to see any messages, and that section B will be masked from view. If Section B were an IBM SIG and Section C a Kaypro SIG, to prevent users who are not members of those groups from seeing any messages in those sections, use the following:

item = R, "R: Read Messages", -, [2:-BC], 11

Note that even with Read access to these sections, a user will not be able to see the messages. He must choose an option from the Menu which sets his mask to those sections:

item = I, "I: IBM SIG", 10-, [2:B], 104, PCUG-O.MEN
 item = K, "K: Kaypro SIG", 10-, [2:C], 104, KUG-O.MEN

Some Functions can be protected from use by anyone who does not know the SYSOP Password, which you set when you set up your BBS-PC!. Access Level 7 is used for this:

item = A, "A: Apply for Membership", 1-4, [7], 3:10

Only New users will see this, and if they select this option from the Menu, they will be prompted with:

SYSOP Password:

An invalid password will abort the Function and return to the Menu.

One additional item worth mentioning is in several instances BBS-PC! may be queried to display section names as in Functions 0, 1 and a few others which may not be as obvious. The sections/access privileges shown by BBS-PC! depend on the access type used as part of the function. Using access type 6 for example will force a display of all sections since no specific section access is required. You may find it more desirable to use access type 0 for these functions.

Control of Display

Further control of how something is displayed is available following the Item Definition field:

"Description",	Displays text within the quotes.
!	Hides the Item from view.
""	Produces a blank line.
"\"	Used when the Menu is made up of
~	Titles, with Item definitions
	listed separately.

Compare the following two ways of writing a Menu:

```
***** Notebook Main Menu - 08/21/85 *****
menu = BBS-O.MEN, "Command (? for menu): ", 1
item = ?, !, -, 6, 101
```

```

title = "+-----The Notebook's Main Menu-----+"
item = B, " B: Bulletins/Help C: Chat w/Sysop "-, 6, 104, BULL-O.MEN
item = E, " E: Exit Notebook L: Leave Message ", 10-, 6, 100
item = E, " E: Exit Notebook L: Leave Message ", -9, 6, 105, EXIT-O.MEN
item = R, " R: Read Messages N: Section Names ", -, 2:-3CE, 11
item = X, " X: Menus on/off #: Note to Sysop ", -, 6, 8
title = "+-----+"
item = C, ""\, -, 6, 7
item = L, ""\, -, 3, 9
item = N, ""\, -, 0, 1
item = #, ""\, -, 6, 104, SYSCOM-O.MEN
endmenu

```

.***** Notebook Main Menu - 08/21/85 *****

```

menu = BBS-O.MEN, "Command (? for menu): ", 1
title = "+-----The Notebook's Main Menu-----+"
title = " B: Bulletins/Help C: Chat w/Sysop "
title = " E: Exit Notebook L: Leave Message "
title = " R: Read Messages N: Section Names "
title = " X: Menus on/off #: Note to Sysop "
title = "+-----+"
item = ?, 1, -, 6, 101
item = B, ""\, -, 6, 104, BULL-O.MEN
item = E, ""\, 10-, 6, 100
item = E, ""\, -9, 6, 105, EXIT-O.MEN
item = R, ""\, -, 2:-3CE, 11
item = X, ""\, -, 6, 8
item = C, ""\, -, 6, 7
item = L, ""\, -, 3, 9
item = N, ""\, -, 0, 1
item = #, ""\, -, 6, 104, SYSCOM-O.MEN
endmenu

```

In the second example, everything displayed is written with a Title Statement, and the Item Definitions follow. There is much to be said for the second method, for it allows you a visual check on the appearance of what is displayed. In the first example (part of an actual Menu used by the author of this manual when he initially set up his system), two lines were out of alignment with the others:

```

+-----The Notebook's Main Menu-----+
| B: Bulletins/Help C: Chat w/Sysop |
| E: Exit Notebook L: Leave Message |
| R: Read Messages N: Section Names |
| X: Menus on/off | #: Note to Sysop |
+-----+

```

Command (? for menu):

It is frustrating to write and compile a Menu only to find one or two things do not look right. Writing Menus as Titles followed by Item Definitions eliminates this problem.

The addition of the backslash to a null display statement ("") suppresses the carriage return at the end of the line. If two Items in the above Menu lacked the backslash, the Menu would appear in this fashion:

The Notebook's Main Menu	
B: Bulletins/Help	C: Chat w/Sysop
E: Exit Notebook	L: Leave Message
R: Read Messages	N: Section Names
X: Menus on/off	#: Note to Sysop

Command (? for menu):

Note that the question mark is not displayed, because it is hidden by the "!" in the Display Description field. This allows you to have number of concealed items which will never be seen by a user.

Because the Menu Display Option (the "1" in the third field of the Menu Statement) instructs BBS-PC! to display the Menu only for Novices, an Expert will see only the prompt:

Command (? for menu):

If an Expert presses Enter, he will see:

Options: BCELRNX#

Entering a "?" will display the full Menu. If a Novice presses Enter, the Menu will be displayed again.

The Endmenu Statement

The last line in the source text for a Menu is the Endmenu Statement:

endmenu

There are no parameters. Without an Endmenu Statement, the compiler would have no way to know it had completed all lines of the Menu source text.

The Comment Statement

If you care to document your source text by including lines that will not be acted upon by the compiler, simply begin them with a period:

.***** Notebook Main Menu - 08/21/85 *****

When the Menu Compiler encounters a period as the first character of a line, it ignores the line and proceeds to the next. In the second example of The Notebook Menu, three lines consist only of a period, followed by a carriage return. This simply separates portions of the Menu visually, making it easier for you to group the Item Statements according to their location on the Menu.

If you create Menus that are as wide as your screen, with three or four Items on one line, doing it with Titles and grouping the Items below is perhaps the best way. Consider the author's first SM-O.MEN (Sysop Maintenance Menu):

The Director's Control Room

The Membership	The System	The Tunnels
A: Add a New Member	B: Display Statistics	1: Back Door to BBS
C: Change Member Record	F: Flush Stale Messages	2: The Phone Booth
D: Delete One Member	M: Check Modem Settings	3: Slip Over to DOS
L: List All Members	N: Change Section Names	4: Go Wait for Call
L/G: List the Guests	R: Reset Bulletin Flags	5: Lending Library
P: Purge the Deadbeats	T: Check System Defaults	6: Open Forum
S: Check User Defaults	V: Change Mode Settings	
U: Update All Users	W: Change Menu Sets	
Y: Caller Log	Z: Edit Terminal Setups	

In this Menu, at least 12 of the Item Statements need to be placed after the Display because only one Item may appear on any line. Placing all Item definitions after the Titles and grouping them by Membership, System, and Tunnels makes it simple to write, and prevents overlooking one. (It is easy to do when you have more than one Function on a line.)

Compiling the Menus

When you have created your source text for your Menus, compiling them is as simple as stated early in this section. Be certain you begin with a Menu Statement and finish with an Endmenu Statement. From the DOS prompt, enter

bbsmenu menus.doc

The compiler will then compile the source text one line at a time, displaying each line on the screen as it does so. Any errors in the source text will cause it to stop compiling and display an error message. (All error messages you may encounter in running BBS-PC! or the Menu Compiler are listed in Appendix ***) The most common errors are:

Syntax errors. Generally they are one of two types (although not limited to two): Either you have neglected to place a comma after each field, or you have used a period instead of a comma. These two account for about 75% of the errors encountered.

Another common error is neglecting to place Display Statements within quotes. The opening quotation mark is usually there, but the closing quote will be missing. Failure to have quotation marks at both beginning and ending of the Display Statement is guaranteed to create havoc during the compiling of a source text.

Another error is attempting to put too many Items and Titles in a Menu. The BBS-PC! Menu Compiler is a limited, special purpose compiler. You are limited to 40 Items and Titles. More than this will cause the compiler to quit with an error message.

When the compiler has finished, it writes the Menu(s) to disk with the .MEN extension. Be sure you run BBS-PC! and check each option on every Menu to make certain they operate as you intended. Incorrect Function Numbers, Access Levels, Display Options, Privilege Levels, etc., will not be corrected by the Menu Compiler. A simple typographical error in entering a Function Number can ruin the best of days if not tested before going on line. Imagine how you would feel if you typed "23" instead of "13" and a caller wanted to Scan the message headers. He would see "User name:" and likely would type in his own name. What he has done instead of scanning the headers was delete himself from the permanent user log!

Creating Graphic Menu Sets

You can use graphics in your menus not only for your own computer, but for any computer for which you have the ASCII special character codes. With a good text editor, this Menu

```
+-- The Hidden Valley Kaypro Club Message Center -----+
!
! 0 = Read Public Bulletins           5 = Kaypro Club Members Only !
! 1 = Post a Public Bulletin          6 = Current Weather Forecast !
! 2 = File Transfer System            7 = Movie Reviews by the Guru !
! 3 = Hidden Valley Helps             8 = Hidden Valley Utilities !
! 4 = Information on Membership        9 = Sign Off for Today      !
```

!----- Contact Ed Jones at 555-1212 for more information -----!

can be converted easily to use ASCII "character graphics" by the search and replace method. The "I" was replaced with ASC(155), and the "-" with ASC(45). The four corners were each replaced separately. If your text editor is not capable of replacing all of them, you can write the source text code to have the compiler insert them, by preceding the code with a backslash ("\"). In the file card menus shown below, the lines with backslashes were written like this:

```

title = "
title = "
title = "92"
title = "
title = "
title = "
title = "92"
title = "
title = "
title = "
title = "
title = "

```

The backslash is ASC(92), and when the Menu Compiler encounters the typed backslash it looks for an ASCII code to follow; when it finds the "92" it prints the backslash.

In designing graphic menus, consider the majority of your users. While most of the author's users are pleased with the appearance of the filecards created with vertical bars, slashes, backslashes and underlines, his Commodore-64 users have a problem, for the underlines appear on their screens as "<-" (left arrows). Except for the login text (which is also a file card), there is an alternate set of Menus for the C-64 users, which simply substitutes hyphens for the underlines.

Unless you are familiar with all the subsets of ASCII special characters as implemented on different systems, the easiest and most considerate way to design alternate Menu sets is to ask one of your users who has a different machine to help you with the various codes.

Introduction to BBS-PC Functions

Functions and Menus

To understand how BBS-PC works, and why it is so flexible, you need to understand what the functions are, and what they do. When a user calls the system, he is presented with one or more "Menus." These menus display a list of options, an example of which might appear as follows:

```
+-----+ The Widget Works Message Center +-----+
| B: Memos to all from The Director                |
| G: Goodbye (Disconnect from System)              |
| R: Read Notices from All Managers                |
| S: Send Daily Report to Your Manager              |
+-----+
```

```
+-----> Sales Department Info <-----+
| 1: Sales Department Memos                        |
| 2: New and Improved Widgets                     |
| 3: Discontinued Widgets                         |
| 4: Notes to Branch Managers                     |
| 5: Notes to Rookie Sales Reps.                  |
+-----+
```

Enter your selection:

The user then enters the letter or number of his choice, and the system responds by performing one or more internal Functions. A "Function" is a set of instructions which tells the computer how to perform certain tasks, i.e., a subroutine. This subroutine has been defined, and is one of a large set of routines within BBS-PC. Although written in a form much like English, it of no use to a computer until it has been compiled. In the same way, Assembly Language is written in a readable form to humans, but the op codes and pseudo-op codes must be assembled to be of use to the computer. Basic is another language which is of little use to a computer, unless it has been processed by an interpreter, which converts it to information understandable to the computer. In much the same way, BBS-PC can be thought of as an interpreter; the main difference is in the type of functions contained within the program. The "Functions" we use here can be thought of as the fundamental "building blocks" of a Bulletin Board System, and much like Basic, these "building blocks" can be put together in many different ways.

The only way to use BBS-PC's Functions is through Menus. You will find that most of your "set-up time" will be spent in developing your own personalized menus. The menus supplied with the program are more for illustration than anything else, although they are complete in a "bare-bones" sense. Micro-Systems Software has developed a "Menu Compiler Language" with which you design your menus, using any text editor (like our own MEDIT) or a word processor that is capable of producing standard ASCII text (such as our MSCRIPT or MSCRIPT-II). When you have finished creating your "source" text, you compile it with the Menu Compiler Program. This compiler program does not actually compile machine readable code, but rather pseudo-code, which the

BBS-PC program interprets and executes. More will be said about this in the section on the Menu Compiler. When BBS-PC is executed, the Functions are called by input from the keyboard when a menu is displayed. Look at the text file for a typical menu:

```
. ***** Widget Works Main Menu - 08/01/85 *****
.
menu = BBS-0.MEN, "Enter your selection (? for menu)", 1
item = ?, 1, 6, -, 101
title = "-----> The Widget Works Message Center <-----"
title = "B: Memos to all from The Director"
title = "G: Goodbye (Disconnect from System)"
title = "R: Read Notices from All Managers"
title = "S: Send Daily Report to Your Manager"
title = "-----"
title = ""
title = "-----> Sales Department Info <-----"
title = "1: Sales Department Memos"
title = "2: New and Improved Widgets"
title = "3: Discontinued Widgets"
title = "4: Notes to Branch Managers"
title = "5: Notes to Rookie Sales Reps."
title = "-----"
.
item = B, "", -, 6, 106, BULL-0.TXT
item = G, "", -, 6, 100
item = R, "", -, 6, 11
item = S, "", 10-, 6, 9
.
item = 1, "", -, 6, 106, SALES-0.TXT
item = 2, "", -, 6, 106, WIDGET-0.NEW
item = 3, "", 10-255, 6, 104, DISCO.MEN
item = 4, "", 100-, 6, 106, MGR-0.TXT
item = 5, "", 5-9, 2:A, 11
endmenu
```

As you can see, Menus are mostly "titles" and "items," both of which contain a number of options and parameters. A comprehensive discussion of menu construction will be found in the section on the Menu Compiler.

Selecting a menu option with the letter or number causes a Function to be executed. "item = R" specifies the the character input from the keyboard; "" states nothing will be displayed, "" suppresses the carriage return after the "; "-" indicates this is available to all users without regard to their privilege level; "11" is the Function to be executed. Entering an "R" (upper or lower case) at the Main Menu causes Function 11 (Read messages) to be executed. Entering a "3" executes Function 104 (Call another menu), which displays the menu DISCO.MEN, but only if the caller's privilege level is within the range 10-255:

```
+-----Discontinued Items to be Pushed-----+
| 1: Obsolete Computer Widgets                |
| 2: Last Week's Pancake Widgets              |
| 3: Outmoded but Functional Widgets          |
| 4: Purchaser's Goofs (real dogs!)           |
+-----+-----+-----+-----+-----+-----+
```

Option 1 on this menu could be defined as follows:

```
title = "1: Obsolete Computer Widgets"
item = 1, "M", -, 6, 106, OBSOLETE.TXT
```

Entering a "1" at this menu would execute Function 106 (CLS, type a file), which clears the screen and displays a text file called OBSOLETE.TXT.

The characters to be input are not limited to one definition throughout the program; each menu defines its own valid input characters (items). Thus every menu could use the number "1" or the letter "R" to mean entirely different things. This is one of the reasons why BBS-PC is so versatile.

This section will detail what the Functions are, and what they do. An example of each is shown, including the different options on input. The letters used in the examples are for illustration only. "P" as shown below to read Private E-Mail could as well be "M" and "L" to Leave a message could as well be "E" for Enter a message. You are not limited to any particular input characters.

The terms New User, Guest and Member differentiate between callers who have not registered for membership, those who have registered but have not yet been admitted, and members. There can be several types of SYSOPS, or System Operators. The term Wizard SYSOP means the person who is the Chief System Operator. Other persons could be Section SYSOPS, and have control over one or more of the sixteen sections of the board, with the ability to determine who has access to that section, what messages should be deleted or moved to another section, which files should be available to the members of that section, and certain other powerful functions. He could also be responsible for maintaining bulletins in that section. A Junior SYSOP is much like a Section SYSOP, but whose control within a section is usually limited to messages and files, but not members. It is not necessary to have more than one SYSOP, but if your needs or desires are such that additional SYSOPS would make the functioning of your board easier and more efficient, you have the power to do so. Perhaps the most common use of Section SYSOPS is to have them totally responsible for maintaining SIGs (Special Interest Groups) on your board. In the Widget Company example above, one person could be Section SYSOP for the Sales Department, while another would be in charge of the Section concerning Discontinued Widgets.

Everyone has the status of "Novice" when first calling the system, or when added as a Member. Only the SYSOP initially carries "Expert" status. A Novice will see all Menus displayed at all times. An Expert is a caller who is familiar enough with the operation of the system he no longer needs the full Menu, and sees only the prompt asking him for a selection. If an Expert presses ENTER at the prompt "Enter your selection:" he would see "Options:

BGRS12345" If a Novice presses ENTER, he sees the full Menu displayed again. There are additional helps throughout the system for Novices. For example, if there is a list of subcommands following a Function (as in the Read messages, or Leave message routines), a Novice will automatically see the list of subcommand options displayed. Anyone, even a first-time caller, could have the ability to change his status while using the BBS.

What New Users, Guests, Members, and SYSOPS are able to do on the board is determined by the privilege level you assign them. Simply determine what you want your users to be able to do, and assign the correct privilege to each. Those Functions which should be under control of the Chief SYSOP are noted below as "Wizard Functions." There MUST be at least one Wizard SYSOP with a privilege of 255; on the disks supplied with BBS-PC, "SYSOP" is the only account listed. This account must remain in the membership records, and the privilege level must be 255 in order to perform any operations directly from the SYSOP Maintenance Menu.

BBS-PC! FUNCTIONS

For purposes of illustration in this manual, we will arbitrarily use the following privileges:

- 0 = Locked from use of the Board (no access at all).
- 1 = New User
- 5 = Guest
- 10 = Member
- 100 = Section SYSOP
- 255 = Wizard SYSOP

Except for the 0 privilege, these levels are entirely arbitrary. You could set them as 1 for New User, 2 for Member, 3 for Section SYSOP, etc. At least one Member MUST have level 255, Wizard SYSOP. The account for SYSOP was included on the master disk for the purpose of initially setting up the board, and should remain. In the examples shown in this section, the access level is used only as an example; you could choose different levels for any reason. Those Functions normally reserved for SYSOPS are so indicated in the applications portion of the Function discussion.

The BBS-PC! Functions are classified in five categories:

- 1) General
- 2) SIG (Special Interest Groups)
- 3) Maintenance
- 4) Special
- 5) Control

As we examine each of them, we will show examples of how they appear in menus, all valid parameters (options) used with each, and what they do. If any Function produces a "Subcommand" which carries a second series of options, these will be shown. Remember that the letters chosen here for illustration of an items are strictly arbitrary. We have used letters that have some connection with the Function itself, such as "E" for "Exit," "N" for "Section Names," and so on. We could have chosen "G" for "Goodbye" and "D" for "Departmental Notices," if that is more in keeping with the theme of the board. What is important is not the letter you choose for the option, but the Function number; the choice at any menu is up to you.

The access level portion of each item statement controls who may use the Function. There are eight types of access:

Item access options:

- 0 - requires specified section access
- 1 - requires specified access w/mask
- 2 - requires RD access w/mask

- 3 - requires WR access w/mask
- 4 - requires DN access w/mask
- 5 - requires UP access w/mask
- 6 - no section access required
- 7 - no section access, requires SYSOP password

These access levels are discussed in the previous manual section (where we are discussing the menu compiler), and will become clear as we demonstrate the Functions. Remember that the statement consists of an item (item = S), the display statement (""), the privilege level (5-), the access level (6[:x]), and the Function number (0[, x]). [:x] and [, x] are optional parameters. Some Functions require additional parameters, such as the name of a menu, or a text file, with others the additional parameters is optional.

Function 0 - User statistics

Parameters used:

None

Description:

Displays automatically on logon, and informs the caller immediately of the following statistics:

Last call : 28-JUL-85
High msg : 9837
Access : 012689AC
Calls : 37
Msgs : 22
Downloads : 55
Uploads : 3
Time limit: 45
Privilege : 10

Application:

In addition to being displayed when the caller logs onto the system, this could be available as a menu option as well. As such, it does not permit the caller to edit any of the items here. If he wishes to make some changes, he would use Function 2 (Change section mask), and/or Function 4 (Change/examine user record). Many users are interested in knowing how often they have called, how many files they have downloaded or uploaded, whether their time limits have been increased, etc. If you have set an Upload Reward, a caller who frequently uploads files to you will be able to see immediately how much additional time he has been given.

Example(s):

item = S, "S: Your Statistics", -, 6, 0

In this example, anyone calling the system could select the option. If you desire, it may be restricted to Members by changing the privilege level:

item = S, "S: Your Statistics", 10-, 6, 0

Function 1 - Section names

Parameters used:

None.

Description:

This function displays the names of active sections for the user, as well as Read, Write, Upload and Download ability in each section. It will appear on the menu for callers with any privilege, and will produce a display similar to the one below. In this example, the caller has access only to sections 0, 1, 7, 8, 9, A, and C:

Sections:

0: [R---] Guest Information
1: [RW-D] General Mail
7: [RWUD] Honda Tune-Up Tips
8: [RW--] Meet-A-Mate
9: [RWUD] Items for Sale
A: [RW-D] Inner-Office Memos
C: [-W--] Memos to Director

After the section number, the user can see what he is able to do within that section:

R = Read open messages in section
W = Write messages in section
U = Upload files to section
D = Download files from section

Sections which cannot be accessed by the user will not appear in the list if access type 0 is used. What each user is able to do in each of his valid sections is determined by the SYSOP.

Application:

Use of this Function allows a user to see which sections are valid ones for him. Many users will experiment with Function 3 (Change section mask) and remove sections from view. This allows them to check what is valid, and if they find any with [----], they should know that it has been masked out. By using Function 3 (Change section mask) they are able to re-enable those sections.

Example(s):

item = N, "N: Section Names", -, 0, 1

THIS PAGE INTENTIONALLY LEFT BLANK

Function 2 - Change section mask

Parameters used:

Any or all section numbers, !, +, -

Description:

Permits a caller to enable or disable sections to avoid reading unwanted material. He may then select only those sections he desires. When he selects this option, he will see the sections displayed, with the addition of a prompt:

Sections:

0: [R—] Guest Information
1: [RW-D] General Mail
2: [RWUD] IBM-PC/XT/AT/jr
6: [RWUD] Compaq Club
9: [RWUD] Items for Sale
A: [RW-D] Inner-Office Memos
C: [-W--] Memos to Director

Section(s):

The user selects the sections wanted by entering the section numbers. If he enters "12" all messages and files in sections 0,6,9,A,C will be hidden from view. If he uses this option now, he will see:

Sections:

0: [—] Guest Information
1: [RW-D] General Mail
2: [RWUD] IBM-PC/XT/AT/jr
6: [—] Compaq Club
9: [—] Items for Sale
A: [—] Inner-Office Memos
C: [—] Memos to Director

Section(s):

This tells him he has access to more sections than he is able to see with this mask set. Entering a "+" or "-" will add or remove sections. He could enter "+6-1" and the M option will show:

Sections:

0: [—] Guest Information
1: [—] General Mail
2: [RWUD] IBM-PC/XT/AT/jr

6: [RWUD] Compaq Club
9: [—] Items for Sale
A: [—] Inner-Office Memos
C: [—] Memos to Director

Section(s):

Simply pressing ENTER or entering "!" will again cause all sections to be enabled. Note that the mask may be entered with the command, as "M+6-1".

Application:

This function is normally used when a caller wishes to save time in reading only those messages in which he has a particular interest. You could have, for example, two sections available to everyone, called "Honda Tune-Up Tips," and "Meet-A-Mate." A happily-married user of this system who drives a Pontiac might have no use for sections 7 and 8, and is able to hide them from view by masking them out. He will not even be aware of them as he will never see them. As SYSOP, you could remove those sections from his access, but if he should desire to have them reinstated, it would require you to do it manually. Allowing the user to set his own section mask on valid sections frees you from one more tedious task.

Example(s):

item = M, "M: Mask for desired sections", 10-, 0, 2

Function 3 - Register/add to membership file

Parameters used:

None

Description:

This Function prompts a caller for his name, location, phone number, password, terminal type, and transfer protocol. Once entered, the information is retained in the permanent user file, and the caller is then designated a "Guest." When this Function is selected, a text file (REGISTER.TXT) is displayed prior to the prompts:

This is the text of REGISTER.TXT. To become a member of this system, you must leave all the information requested below. When you log off, you will be asked if you want to leave a comment. At that time, you MUST give us your membership number in the Pahokee-Loxahatchee Users Group for access to those sections.

Your name :
City, State :
Phone number:
Password :

Terminals:

0: ASCII Terminal 80 x 24
1: IBM PC/XT/AT/jr
2: TRS-80 I/III
3: Apple 80-col
4: Apple 40-col
5: TI 700
6: Commodore 64
7: Televideo 925

Terminal type:**Options:**

T: Standard ASCII text
X: Standard XMODEM
C: XMODEM with CRC
S: Smartcom II

Protocol (T X C S):

After all questions have been answered, the caller is added to the permanent user file. Should he attempt to use this Function again, he will be informed that he is already in the file. (No duplicate names are permitted.)

Application:

In a closed system, this Function promotes a "New User" to "Guest" status. The only difference is that New Users are not recorded in the permanent user files. In an open system, this can be used to promote a "New User" to "Member" status. This frees the SYSOP from entering the information for everyone who joins the system, and provides a permanent record of his location and phone number. SYSOPs frequently need to contact a user by voice phone, and the inclusion of the phone number in the record provides a convenient way to keep the information at hand.

Example(s):

item = A, "A: Apply for Membership", 1, 6, 3:5

Appended to the Function number is the privilege level which will be assigned the caller immediately upon completing the function. If you want an open system in which New Users are able to add themselves as Members, you could use the following:

item = J, "J: Join the system as a Member", 1, 6, 3:10

This immediately give him full Member privileges, including all sections used for Members.

Function 4 - Change/examine user record

Parameters used:

None.

Description:

This Function permits the caller to change certain information in his permanent user record, and displays the following:

John Caller
Indiantown, FL

1: Terminal : ASCII Text
2: Section : 1
3: Password : HAMMERHEAD
4: Status : Novice
5: Protocol : Xmodem
6: CLS codes : 12 0 0
7: BS codes : 8 0 0
8: Page size : 80 x 24
9: Linefeeds : Yes
0: Nulls : 0

Enter line to change:

Application:

Selecting this Function from the Menu allows the user to change any of the above information in his permanent record. Users frequently change terminals or software, and this allows them to make the necessary changes to their records, freeing the SYSOP from having to perform that chore. Selecting each of the options will display the information in the right-hand column. As each option is exercised, the system will again display the table above, and the prompt ("Enter line to change:"). After all desired changes have been made, pressing ENTER will make the changes in the permanent record and return to the menu from which this Function was selected.

(Option)	(Result)
1: Terminal	Terminals:
	0: ASCII Terminal 80 x 24
	1: IBM PC/XT/AT/jr
	2: TRS-80 I/III
	3: Apple 80-col
	4: Apple 40-col
	5: TI 700

6: Commodore 64
7: Televideo 925

Terminal type:

He then enters the number of his terminal, and his records now reflect the defaults set for that terminal.

2: Section

Sections:

0: [R---] Guest Information
1: [RW-D] General Mail
2: [RWUD] IBM-PC/XT/AT/jr
6: [RWUD] Compaq Club
9: [RWUD] Items for Sale
A: [RW-D] Inner-Office Memos
C: [-W---] Memos to Director

Default section:

The section chosen as the default section is where messages will be stored if no section number is specified when a message is saved.

3: Password

Password:

The caller enters his new password, and it is stored with his permanent record.

4: Status

Expert user (Y/N):

When a user exercises this option to switch full menu displays on or off, his permanent record is changed, unlike using Function 8 (Expert menu toggle), which changes it only for the duration of the current call.

Options 5-0 allow the user to make permanent changes in his terminal configuration:

5: Protocol

Options:

T: ASCII text
X: Standard XMODEM
C: XMODEM with CRC
S: Smartcom II

Protocol (T X C S):

6: CLS codes

CLS code1:
CLS code2:
CLS code3:

7: BS codes BS code1:
 BS code2:
 BS code3:

Note that in options 6; the most common backspace code is ASC(8), and in option 7, clear screen code is ASC(12). It is not necessary to enter zeros for codes 2 and 3; simply press ENTER and continue editing the record.

8: Page size Page width:
 Page len :

The page width selected is usually the maximum width of the screen display, although it could be shorter (or longer). Page length is the number of lines that will be displayed on the screen before pausing and displaying the "- more (Y/N/NS)?" prompt. You should advise your users that if they want a continuous display without the prompt, they should set the page length to zero (0).

9: Linefeeds Linefeeds (Y/N):

0: Nulls Number of NULs:

Some systems are incapable of keeping up with the rate of transmission from BBS-PC!. This is especially true at 1200 and 2400 baud, and the user may choose this option to slow down the transmission. He may select any number of NULs from 0 to 255. (At 1200 baud, *** NULs is equivalent to *** seconds.) If a caller wants to dump all incoming information to his printer, and does not have a large buffer on the printer, some NULs are usually necessary to prevent loss of data.

Example(s):

item = C, "C: Change your Record", 10-, 6, 4

In this example, use of the Function is limited to Members.

Function 5 - Print caller log

Parameters used:None

Description:

Displays information on recent callers: Which node was used, the user's parameters and baud rate, time in and out of the system, what type of disconnect, and what he did while on the system:

#13179 - Node #1 300:7E1
James Hargraves
East Haverhill, CT
In : 10:42 AM 10-AUG-85
Out: 10:56 AM <Sleep Disconnect>
Sum: 4 Downloads, 1 Upload

#13178 - Node #1 1200:8N1
Steven Grant
Los Angeles, CA
In : 10:16 AM 10-AUG-85
Out: 10:31 AM
Sum: 2 Messages. 1 Download

#13177 - Node #1 LOCAL
Karl Meyer
West Palm Beach, FL
In : 09:02 AM 10-AUG-85
Out: 09:06 AM
Sum: 1 Message

#13176 - Node #1 1200:7E1
Terry Richter
Boynton Beach, FL
In : 08:03 AM 10-AUG-85
Out: 08:58 AM <Time Exceeded>
Sum: Snoop

#13175 - Node #1 1200:8N1
Jim Anderson
Billings, MT
In : 04:14 AM 10-AUG-85
Out: 04:42 AM
Sum: 1 Message, 1 Upload, 2 Downloads

Application:

Many systems restrict use of this Function to Members, to frustrate the troublemakers who simply want to see who has been using the system, and attempt to find the caller's password to leave bogus messages. In the first example below, anyone may view the caller log; in the second, it is restricted to members. The names shown are only those within the specified range of privileges you established it when setting up the system. SYSOPS, however, will see all names, regardless of privilege range.

Example(s):

```
item = C, "C: Print Caller List", -, 6, 5
item = U, "U: Print User Log", 10-, 6, 5
```

Function 6 - Time on system

Parameters used:

None

Description:

This Function displays the login time, date, current time, and elapsed time. If the amount of time remaining on this call (or day, depending on how you have established it) is two hours or less, it will also display the time remaining:

Current : 10:12 PM 17-AUG-85

Logged : 09:45 PM 17-AUG-85

Elapsed : 00:27

32 minute(s) remaining

Application:

Many users not only like to know how much time they've used, they need to keep track of time if they are calling long distance or using a subscription long distance service. By selecting this Function, they are kept aware of the time they have used. If you charge for time on system, the caller is able to prevent running up too high a bill.

Example(s):

item = T, "T: Time You've Been Here", -, 6, 6

Function 7 - Chat with SYSOP

Parameters used:

None

Description:

This Function initiates the direct communication mode between the caller and the SYSOP (or whoever is sitting at the BBS terminal).

Application:

If the Chat has been enabled, this function sends a signal to the speaker and a displays, "Paging SYSOP . . ." and a series of dots to the screen. (The audible signal will be heard only if the Local Bell is turned on at the Node defaults.) The SYSOP may then respond to the call by pressing the space bar (or any key), and enter the chat mode to intercommunicate with the caller. If the SYSOP ignores the summons, a message will be displayed on the screen saying the SYSOP is not available at this time. If the Chat has not been enabled, this function displays a message stating "SYSOP not available." If the Chat is disabled, or when Chat is terminated, the system returns to whatever it was doing at the time Chat was begun. Even if the Chat mode is turned off, the SYSOP may engage the caller in chat with the F1 key.

Example(s):

item = C, "C: Chat with The Director", -, 6, 7

Function 8 - Expert menu toggle

Parameters used:

None

Description:

Allows the caller to temporarily toggle his status from expert to novice. User status can be changed permanently with Function 4.

Application:

A "Novice" menu is a display of the entire menu. Users should be given the option of using the Expert menus to eliminate the time needed for display. When the full menu display is turned off, it affects all menus on the system, unless you have set them to display at all times. This is a "Novice" menu:

```
←→ The Widget Works Message Center ←→  
| B: Memos to all from The Director |  
| G: Goodbye (Disconnect from System)|  
| R: Read Notices from All Managers |  
| S: Send Daily Report to Your Manager|  
←→
```

Command :

An Expert will see only the prompt. If he enters a "?" he will see the full Menu displayed. Simply pressing ENTER shows:

Options: BGRS

As a menu option, this will switch from one set to another for the duration of the current call. It can be changed permanently by using Function 4 (Change/examine user record). All New Users are set to "Novice" mode.

Example(s):

item = X, "Expert Menu Switch", -, 6, 8

Function 9 - Leave a message

Parameters used:

None. See parameters used with Storing the message below.

Description:

Selecting this function initiates the routines to post messages on the system. Once the message has been entered, the user is then prompted with a number of options which are detailed below.

Example(s):

(a) item = L, "L: Leave a Message", 10-, 6, 9

Application:

When the caller selects the Function to Leave a message he is prompted with:

To:

After the name of the addressee is entered, the caller will next be prompted with:

Subj:

Enter the subject of the message here. Later, any replies to this message will carry this same subject along with a reference to the message number.

The message's "From:" field will be filled in with the sender's name. When both the "To:" and "Subj:" fields have been entered, if the user is a Novice, the system responds with:

Enter message text.

Carriage return on blank line to end.

Expert users will see only the next prompt

1:

Users at this point may type the text of a message. BBS-PC features a unique word wrap function, much like a word processor. Whenever a word is typed that is too large large to fit on the current line, BBS-PC will erase however much of the word entered on the current line, re-enter it on the next line, and continue. Pressing <ENTER> is only required to end a paragraph or terminated message input.

Please note that there are two items that need to be set correctly for this function to perform properly. First, a callers terminal must perform a destructive backspace (when the cursor is backed up, the character is erased). Since each terminal performs this differently, BBS-PC allows callers to set BS (backspace) codes as part of their permanent record. Even in the worst case, it has been our experience that setting these codes to 8 32 8 (a backspace followed by a space and another backspace) will work correctly.

Second, the page width parameter must be correct for the callers display width. Unless BBS-PC knows where the end of a line is, it cannot hope to wrap the words properly.

To cause a blank line to appear in the message (usually desired between paragraphs), a period is typed on a line all by itself. Also, any line that begins with a space or a tab character will start on a new line on the caller's screen.

Let's take an example. The message

```
1: John -  
2:      Terry and I will be at your place  
3: around 9 tonight. We will be in either my  
4: Ford or his dad's Honda. Look for us then.  
5:      Mark
```

This will be reformatted, when read, to appear as

```
John -  
      Terry and I will be at your place around 9  
tonight. We will be in either my Ford or his dad's  
Honda. Look for us then.  
      Mark
```

To add blank lines between the paragraphs

```
1: John -  
2: .  
3:      Terry and I will be at your place  
3: around 9 tonight. We will be in either my  
4: Ford or his dad's Honda. Look for us then.  
6: .  
7:      Mark
```

(Note the periods on lines 2 and 6.) This will now appear as

John -

```
      Terry and I will be at your place around 9  
tonight. We will be in either my Ford or his dad's
```

Honda. Look for us then.

Mark

To create a message with a series of short lines that all start at the left hand side of the caller's screen, but do not begin with a space or tab, a provision has been made to start each line with a period. For example, the "bordered" message:

```
1: .*****
2: .*  FOR SALE      *
3: .*   1 IBM PC     *
4: .*   $500         *
5: .*****
```

This message will be displayed in the box. Without the periods, each line would be re-formatted, and it would ruin the effect of the border. Another option is to store the message as "unformatted". Unformatted messages are not required to have periods at the start of each line and in fact if entered will be shown as part of the message when viewed. With BBS-PC's message re-formatting, it is usually not necessary to force a message to have short lines. It is only necessary to get each caller to set the CON width parameter correctly for their display size. For example:

```
1: John -
2: .
3: .Please
4: .stop
5: .leaving
6: .hard-formatted
7: .messages!
8: .
9:  It looks pretty awful to people
10: with nice wide 80 column CRTS. OK?
11: .
12:           Mark
```

When a caller has completed typing in a message, pressing <ENTER> on a new line will end the message entry and BBS-PC shows the following:

Subcommand:

This is the prompt for the message leave subcommand mode. Several options are now available, and may be seen by entering a "?" (Novices see this automatically.)

Valid subcommands:

A: Abort message
C: Continue message
D: Delete line

E: Edit line
H: Additional help
I: Insert line
L: List message
N: Section names
P: Preview message
R: Replace line
S: Store message

These options at the subcommand prompt are:

A: Abort message - permits you to cancel the Leave Function. A prompt will ask "Are you sure (Y/N)?" in case the A had been entered in error.

C: Continue message - allows you to add more text to the message before storing it.

D: Delete line - allows you to delete a line of the message you just entered. Entering "D" will display a prompt asking which line; you may also enter "D#", where "#" is the line number you wish deleted. The line will be displayed, and you will be prompted for verification. If you respond "Y", the line will be removed. All subsequent lines will be renumbered.

E: Edit line - allows you to edit a line. Enter "E#", where "#" is the line you wish to edit. That line will be displayed, and you will be asked for the string you wish to replace. It is not necessary to retype the entire line to correct a simple error. Enter whatever characters you wish to modify. Note that you must enter enough characters to make the string unique. The FIRST occurrence of the string is the one that shall be replaced. If the search string cannot be located in the line, the edit will be aborted. If the search string is located, you will be prompted for the replacement.

Enter the new characters to change the string, or press RETURN to delete the search string completely. After the replacement is completed, you will see the edited line and asked if it is correct. A "Y" replaces the former line with the edited line; any other response will abort the edit.

L: List message - displays the message line by line, exactly as entered. You may enter "L" by itself and list the entire message, or "L#", where "#" is the beginning line number, and list only part of the message.

N: Section names - displays current section names. This allows you to review your valid sections if you wish to store the message in a section other than the default section.

P: Preview message - displays the message in formatted form. It will display the text of the message formatted to your current page size. It can also display the text formatted to ANY size screen; enter "P40" and see the text formatted to a screen of 40-column width. The minimum width is 32 columns, the maximum is 132.

R: Replace line - replaces an entire line in the message. Enter "R#", where "#" is the number of the line to be replaced. The current line is displayed, and you are prompted to enter the new line. After the new line has been typed, pressing ENTER replaces the old line with the new one.

S: Store message - will save the message to the disk. There are a number of sub-options with the Store message option:

S = Store as open message in default section (or current masked section).

S[L] = Store as open message, locked to prevent scrolling off after maximum messages entered, or during purge of old messages. (Only a SYSOP can lock a message.)

S[P] = Store as private message in default section.

S[x] = Store as open message in a particular section, where "x" is the section number.

S[Pr] = Store as private message in a specific section.

S[U] = Store as open unformatted message. Unformatted messages will not be reformatted by the system, and will appear exactly as they were entered, regardless of line length.

S[PU] = Store as unformatted private message.

S[PUx] = Store as unformatted private message in a particular section.

NOTE: In all cases, the section number (if used) must follow all other options.

Application:

Function 10 - Leave message TO SYSOP

Parameters used:

None.

Description:

This Function allows a caller to leave a private comment to the SYSOP, providing you have set aside a section for that purpose.

Application:

This is much the same as the Leave Message Function (9), except that the user is not prompted for an addressee or a subject. In the example below, the access level "3:C" sets a mask to section C. This assumes you have reserved section C for "Comments to Sysop". and have set the access to [-W--]. The message will be stored in that section, and only SYSOPS will be able to read it, and then only if they have Read access to that section [R---].

Example(s):

item = C, "C: Comment to SYSOP", -, 3:C, 10

Function 11 - Read messages

Parameters used:

F, I, M, N, R, S, T

Description:

This Function initiates the Read messages routine. Entering "R" without one of the optional parameters will display the subcommand prompt. Novices will see the following automatically; Expert users enter a "?" to see the list of options:

Options:

A: Abort command
F: Forward multiple
I: Individual messages
M: Marked messages
N: New messages
R: Reverse multiple
S: Selective messages
T: Follow threads

Application:

The optional parameters F, I, R, S, and T may have the beginning numbers appended when entered with the "R" (as RF6752). If the beginning number is not appended, the user will be prompted for it. The N option (New Messages) functions the same as a starting number, since it takes its value from the variable stored with the user's permanent record as the high message counter; it operates only on forward retrieval.

A: Abort command - lets you return to the main command prompt.

F: Forward multiple - displays a notice telling you how many messages are currently in file, and what the first and last message numbers are. Then it will prompt you for the number where you want to begin reading. You may also enter N to read all new messages since you last used the system. "RN" is one of the most common ways to use the Read message option.

I: Individual messages - displays lowest and highest numbers and asks you for a message number. After reading that message, you will again be prompted for a message number. Pressing ENTER will return to the main menu prompt.

M: Marked messages - calls any messages you have waiting for you when you log on. It will also display messages you have marked for reading while scanning the headers, as explained below in Function 13 (Scan messages).

N: New messages - displays only messages since you last called the system. The "N" is actually a variable that is equal to the highest message you have read, plus one. Thus, if your permanent record shows the highest message to be 3471, using "N" in any Read command will begin with 3472. (Entering "RN" is the same as entering "RFN.")

R: Reverse multiple - functions exactly as the "F" subcommand does, except for direction.

S: Selective messages - allows you to search for a specific string of characters in the To, From or Subject fields of the message header.

T: Follow threads - allows you to follow the message threads. A message thread occurs when users read mail and then use the reply option to add an additional message on the same subject. These messages, as well as any other replies, are linked together to form a thread. Any time you read a message, if there is a message number in the subject field, that is an indication of the fact that this message is a reply to another message. If there are later replies to the message you are reading, these will be listed after the message text is displayed. In either case, a thread exists.

There are three options available from the read message subcommand prompt that deal with message threads, but if you already know that whenever you happen upon a thread, you want to read all subsequent replies before reading the next unrelated message; this is the subcommand to use. Entering "RTN" will read all new messages and follow threads, "RF100" will read forward from message #100 and follow threads, and whenever a message has a reply, and you tell the system to continue to the next message, the reply will be shown next.

Once all replies are read, including replies to replies, you will return to your former spot in the message base and continue with the next message in the file. Any messages that you "read ahead" with this subcommand will not be re-read as you move forward through the message base. Also note that if you read message #100 and its reply, message #200, and log off, your high message is set to 200, since that is the highest message read. When you call back, you will have to use "RF101" to see messages between 100 and 200. This is entirely normal.

When you reply to a message you have read, while reading threads, and continue, you will see the reply you just left. This is also entirely normal, since that message now makes up the next message in the thread.

The message display is always the same. In other words, no matter how you are reading the mail (forward, reverse, individual, or new), each time a message is shown, it will appear the same way.

First there is the message header, which tells you the message number, what section it is from, the name of the sender, the name of the person it was sent to, and the subject. Also, there are a series of flags that may or may not appear next to the name of the addressee, depending upon what action has been taken about the message. In addition, if the message is a reply to an earlier message, the message number of the first message will be stored in

the subject of the second, giving you the ability to trace back conversations. Any replies to the current message are listed following the message text display. A sample message header might look like this:

```
Msg: #100  Sec: 1 - General Mail
Date: 23-JAN-85 02:53 PM
To: John Doe (X) (R) (P) (F)
From: Joe Public
Subj: #35 - Hot dogs and hamburgers
```

The first line tells you the message number and what section it is from. The second line tells you what date and time it was left on, assuming that the system clock was set correctly (clock/calendar boards are excellent investments for SYSOPs!). The third line tells you to whom the message was sent. Following that name are the flags. They are all present here in the example, but do not have to all occur in real life. They are:

Flag Meaning

- (X) The person to whom the message was sent has read the message. Helpful in determining which messages may be deleted.
- (R) Someone, not necessarily the person to whom it was addressed replied to the message with the RE subcommand.
- (P) The message was stored as personal. Only the person to whom it was addressed and the SYSOPs can read it.
- (F) The message has been forwarded from another section by a SYSOP. Usually, this occurs when someone either leaves a message in the wrong section or leaves a message in the non-member section that you want the member to see.
- (L) Message has been locked from rolling off the system. Only SYSOPs can roll lock a message.
- (U) The message was stored as unformatted. No attempt is made to reformat message that have been stored this way.

The fourth line tells you who sent the message. The fifth and last line gives you the subject of this message. In our example, you can see that there is the number of a prior message. To see what is being discussed, you may note this number and refer to it later. This message, personal, forwarded, read, and replied to, is in itself a reply to an earlier message.

Following the message header is the actual message text. The caller who is reading the message can pause output with CTL-S, resume with any key, or abort with CTL-P. When the display of the actual message is through, the message read subcommand prompt will appear. It appears as:

(UA RE T) (D F):

If you type a "?" and press RETURN, you will see the options, like so:

Options:

C : Continue
M : Mark
NS: No stopping
R : Read message
RE: Reply
RP: Read previous
RT: Read thread
T : Terminate
UA: Use address
D : Delete
F : Forward

Option C will continue and either display the next message, or return to the command level (depending on whether or not you are in a multiple message retrieval mode). Pressing RETURN all by itself defaults to this option.

Option NS will being displaying messages, assuming you are retrieving multiple messages, and not pause at this prompt again. You may pause the message stream with CTL-S and resume with any other key. When reading messages non-stop, CTL-P will cause the prompt to re-appear.

Option R will display the message text again.

Option RE will enter a message addressed to the person who left the message you are reading, and using the same subject and section number. You can't reply to a message that is in a section where you do not have write access. BBS-PC must save the message in the same section to preserve the continuity of the subject.

Option RP only functions when the message you are reading is a reply to another message. This is indicated by the message number of the previous message appearing in the subject field of this message. If this is the case, entering "RP" will cause BBS-PC to display the previous message and allow you to perform any of the normal message read functions. When you continue from that point, you will proceed to the next message following the original message you read (i.e. you return to where you were in the message base).

Option RR only functions when there are subsequent replies to the message you are reading. These will be listed immediately following the text of the message. When you enter "RR" at the subcommand prompt, the first reply will be displayed. Pressing <RETURN> to continue reading will display the rest of the replies and then continue with the message following the original. Any messages read will not be re-read later during the same read operation. Using the RR option differs slightly than using the "T" subcommand of the main "R" command (follow threads). When following the thread, any replies to replies will be read automatically. When using the RR option, if one of the replies has replies of its own, you must enter "RR" once again to read those replies.

Option T will abort the message retrieval mode and return to the main command prompt.

Option UA will allow you to leave a message, automatically addressing it to the person who left the message you just read, but unlike RE, it will ask you for a new subject. Also, there is no section default here. This option is identical to using the "L" command, save that it addresses the "To" field for you, saving time and effort. You would use this option when you read a message from someone and decide you wish to communicate on an unrelated topic. UA also will not cause the (R) flag to appear.

Option D, if it appears, means that you can delete the message. This would indicate that either you left the message, it was left to you, or you are a SYSOP. If you have the D option displayed, you may delete the message.

Option F, for SYSOPs only, lets you forward a message to a later spot in the message base. The message is actually deleted from the current location and inserted into the new slot. This will place it in front of the new messages so that users could not avoid a forwarded message by accident. When you select "F", you will be prompted for the new section number, the new "To" field, the new subject, and whether you want it stored personal or not. If you press RETURN by itself at any of these prompts, the old information will be preserved. Otherwise, you may enter the new data. The only field you cannot change in the message header is the name of the sender.
Example(s):

item = R, "R: Read Messages", -, 6, 11

Function 12 - Read E-MAIL

Parameters used:

B, F, M, T

Description:

This Function allows a user to read messages that have been stored in a special section set aside for E-Mail. Selecting the item without a parameter displays the following:

Subcommand:

A: Abort command
B: Both To and From you
F: Mail From you
M: Marked mail
T: Mail To you

Any of the parameters except A may be entered with the Item choice, as EM, or EB.

Application:

E-Mail is a popular feature on many Bulletin Boards. By setting one section aside for E-Mail messages and prohibiting private messages in the other sections, your callers will never see "Msg. # 9999 - Private to John Doe." While you could have both private messages in any or all sections and E-Mail, choosing one or the other is the more common method. You should tell your users that the M option (Marked mail) is faster than the T option (Mail To you). The T option forces BBS-PC to search all active messages in the message base to find the caller's name, while the M option finds flagged messages much faster. In the example used here, Section E has been set up as the E-Mail section, and when a caller selects E from the Menu, all other sections are masked out. The user must have Read and Write access to Section E. In the example below, E-Mail is limited to Members.

Example(s):

item = E, "E: Read E-Mail", 10-, 2:E, 12

Function 13 - Scan messages

Parameters used:

F, I, M, N, R, S, T

Description:

This Function permits the caller to view only the message headers without displaying the messages. All options allowed in Function 9 (Read messages) are valid with the Scan message Function. (Scan is the same as Read, except the body of the message is not displayed unless asked for.) The following is displayed:

Msg: #4097 Sec: 1 - Inter-Office Mail
Date: 3-AUG-85 03:27 PM
To: Terry Richter (X) (R) (P) (F) (U)
From: Dan Rostkoski
Subj: #4006 - Adios Airlines

(R M T) (D F):

Which of the above lines are displayed is determined by the way you defined the sections from the SYSOP maintenance menu. The X,R,P,F, and U in the To field are flags which are explained in the Message Base section of the manual. The prompt at the end of the header allows you to:

R = Read message
M = Mark (to read later)
T = Terminate reading and return to the menu
D = Delete message (if it is to or from you, or you are a SYSOP)
F = Forward (if you are a SYSOP)

Like the Read messages Function, the Scan messages Function allows several optional choices to be entered with the S. They are:

S[F],[Fx],[I],[Ix],[M],[N],[R],[Rx],[S],[Sx],[T],[Tx],[TN]

SFx will begin scanning messages forward from a given number (where "x" is the number), RN will scan new messages. Simply entering "S" will display the Subcommand prompt. If the caller enters "?", or if he is a Novice, he will see this list of options here:

Options:
A: Abort command
F: Forward multiple
I: Individual messages

M: Marked messages
N: New messages
R: Reverse multiple
S: Selective messages
T: Follow threads

Let's take each of these subcommands in turn.

A: Abort command lets you return to the main command prompt.

F: Forward multiple displays a notice telling you how many messages are currently in the Message Base, and what the lowest and highest numbers are. Then it will prompt you for the number where you want to begin scanning. You may also enter N to scan all new messages since you last used the system. "SN" is a common way to use this Function.

I: Individual messages displays lowest and highest numbers and asks you for a message number. After scanning that message, you will again be prompted for a message number. Pressing ENTER will return to the main menu prompt.

M: Marked messages calls the headers of any messages you have waiting for you when you log on. It will also display message headers you have marked while scanning the headers.

N: New messages displays the headers only of messages posted since you last called the system.

R: Reverse multiple functions exactly as the "F" subcommand does, except for direction.

S: Selective messages allows you to search for a specific string of characters in the To, From or Subject fields of the message header.

T: Follow threads allows you to scan all replies to a message before going to the next message. See Read message function for more details on this powerful feature.

NOTE: If you read a message while scanning the headers, you will be prompted with all the Read message prompts, and will be able to Reply, Use Address, etc. Even if not displayed, you can always use any of the options.

ACCESS LEVEL:

All.

Example(s):

item = S, "Scan Message Headers", -, 6, 13

Function 14 - Delete a message

Parameters used:

Message number.

Description:

This Function allows Members to delete messages from the system. If the message is not to him or from him, he will be told he cannot delete it. Anyone with SYSOP privileges may delete any messages in any sections to which he has SYSOP status, regardless of sender or recipient. If the command is not followed by the number of the message to delete ("Dxxx"), a prompt will appear asking for it. Pressing ENTER at the prompt without entering a number will return to the main menu prompt.

Example(s):

item = D, "D: Delete messages", 10-, 6, 14

Function 15 - Print file catalog

Parameters used:

/dir, /A, /S, Wildmask

Description:

Entering "C" without any optional parameter displays the catalog of files in the Download area. Only the name, date of entry, length of file and type (B = binary or T = text), and a brief description of each file are displayed, in "natural order" (the order in which they were entered into the catalog). The catalog is displayed in the following form:

```
BOOK CONTRACTS 03-JAN-85 4186-T How to make money.
AMORTIZE.COM    12-JAN-85 12766-B Loan table.
GRAFFIX.BAS     22-MAR-85 87032-B Light pen art prog.
AIRLINE SCHEDS 08-JUN-85 2185-T Adios Airlines.
```

Application:

This Function allows a user to see what files are in the catalog that may be downloaded. When this option is selected without a parameter, the name, date of entry, length of file and type (B = binary or T = text), and a brief description of each file are displayed, in "natural order" (the order in which they were entered into the catalog). The optional parameters are:

C : Displays catalog in natural order (as above).
C/2 : Displays the files in Directory 2 (as above).
C/A : Displays newest files first (aged order).
C/S : Displays catalog in alphabetically sorted order.

You may use a "wildmask" to find particular entries in the Catalog.

CA : Displays only files beginning with the letter A:

```
AMORTIZE.COM    12-JAN-85 12766-B Loan table.
AIRLINE SCHEDS 08-JUN-85 2185-T Adios Airlines.
```

You may use an asterisk ("*") with the wildmask to determine the position of the letter:

C**A : Displays only files with the letter "A" in the third position of the name:

```
GRAFFIX.BAS     22-MAR-85 87032-B Light pen art prog.
```

A "wildmask" search may also specify files by extension:

C*.BAS : Displays all files with the extension .BAS.

If the optional /S is used, the files will be displayed in alphabetically sorted order:

AIRLINE SCHEDS	08-JUN-85	2185-T	Adios Airlines.
AMORTIZE.COM	12-JAN-85	12766-B	Loan table.
BOOK CONTRACTS	03-JAN-85	4186-T	How to make money.
GRAFPX.BAS	22-MAR-85	87032-B	Light pen art prog.

If the optional /A is used, the catalog will be displayed in "aged order," with the newest files listed first. This will reverse the list that is displayed without using any optional parameters:

AIRLINE SCHEDS	08-JUN-85	2185-T	Adios Airlines.
GRAFPX.BAS	22-MAR-85	87032-B	Light pen art prog.
AMORTIZE.COM	12-JAN-85	12766-B	Loan table.
BOOK CONTRACTS	03-JAN-85	4186-T	How to make money.

This is convenient for callers who download files frequently, and who are thus able to see the newest entries listed.

Another way of finding particular files is by using Function 44, Search for string in the description of the file.

Example(s):

item = C, "C: Catalog of Files", -, 4, 15

Function 16 - Browse thru files

Parameters used:

/dir, /A, /S, Wildmask

Description:

This Function displays an extended catalog (header and information on each file) similar to the following:

```
FILE: BOOK CONTRACTS 4186-T
DATE: 03-JAN-85 (L)
DIR: 1 Sec: 0 - General Info
FROM: John Doe
ACC: 13
```

How to make more money.

(D R T) (K M):

The options at the end of this header display are:

```
D: Download the file.
R: Read a formatted text file.
T: Terminate browse files Function.
K: Kill file (for owner w/in 24 hrs, and SYSOPS).
M: Move file to another section/directory (SYSOPS only).
```

Application:

This Function is a convenience for the user, in that you may read or download each file as it appears simply by entering R or D. The files will be displayed in aged order unless otherwise specified. The same options used with the Print file catalog Function may be used with the Browse Function:

```
B      : Displays extended catalog in aged order.
B/2    : Displays extended catalog of directory 2.
B/N    : Displays extended catalog in natural order.
B/S    : Displays extended catalog in alphabetically sorted order.
```

You may use a "wildmask" to find particular entries in the Catalog.

```
BA     : Displays only files beginning with the letter A:
```

You may use an asterisk ("*") with the wildmask to determine the position of the letter:

BA** : Displays only files with the letter "A" in the third position of the name:

Example(s):

item = B, "B: Browse the Catalog", -, 6, 16

Function 17 - Upload a file

Parameters used:

/Protocol, Filename.

Description:

This Function handles the upload routines for callers from remote locations.

Application:

When the caller initiates this Function without specifying a file name, the following prompt is displayed:

File name:

He could also have entered "T filename" at the Menu prompt. After entering the name of the file he is going to upload, he will see the next prompt:

Section (? for list):

If he is not sure of the section in which the file belongs, he enters a question mark, and if his section mask has not been set to a single section, the list of valid upload sections is displayed:

Sections:

- 1: [RWUD] General Mail
- 2: [RWUD] Local Interest
- 3: [RWUD] IBM-PC/XT/AT/jr
- 4: [RWUD] Sales Department
- 5: [RWUD] Managers' Notices
- 6: [RWUD] Widget Catalog
- A: [RWUD] New Salesreps

Section (? for list):

He enters the appropriate section number, and is then prompted for a brief description of what he is uploading to the system:

Description (40 chars max)

[-----5-----10-----15-----20-----25-----30-----35-----]

If he attempts to enter more than the 40-character maximum, a Control-G (Bell) will be transmitted to his terminal with each superfluous character. After the upload is completed, he will see a message telling whether it was successful or not.

If a file already exists with that name, he will receive an error message informing him of the fact, and he should attempt to upload it under a different name.

Example(s):

item = T, "T: Transmit File to System", -, 0, 17

Function 18 - Upload a file from local

Parameters used:

Filename.

Description:

Initiates the upload routines from your local terminal. There is a major difference between this and Function 17, in that when you upload a file locally, the file must reside in one of the paths specified at initialization. As in Function 17, you may enter "U filename" to bypass the first prompt. As SYSOP, you are asked which directory the file belongs in, but the other prompts are the same.

Application:

This Function should be reserved to SYSOPs only.

Example(s):

item = U, "U: Upload a file locally", 150-255, 6, 18

Function 19 - Download a file

Parameters used:

/Protocol, Filename.

Description:

This Function initiates the routines to transmit a file from the system to a remote caller.

Application:

To begin the download, enter "D filename" and follow the prompts. It is worth noting that this area of using the BBS seems to present the most difficulty for inexperienced (and even veteran) users. Given a catalog entry like:

AIRLINE SCHEDS 08-JUN-85 2185-T Adios Airlines.

many people will attempt to enter the name as "08-JUN-85" or "2185-T" or even "Adios Airlines" and seemingly be unaware of the name "AIRLINE SCHEDS." Why this should be so is a mystery, but you would be well advised to put very explicit instructions in your UDHELP.TXT file.

Example(s):

item = D, "D: Download A File", -, 4, 19

Function 20 - Read a file

Parameters used:

Filename.

Description:

This Function permits the user to display a text file which is formatted to his page dimensions. It does not open and close his buffer, nor does it use any of the built-in terminal protocols (Xmodem, SmartCom, etc.).

Application:

Although this Function does not open and close the user's buffer, if his software requires him to do this manually, he may open his buffer, read the file, then close the buffer. Users frequently like to read something before downloading it to be sure it is something they want. Using this Function permits the user to read it first before he fills up his disk space with an unwanted text file.

If he has not entered the name of the file as a parameter (R filename), he will be prompted for it.

Example(s):

item = R, "R: Read a Text File", -, 6, 20

Function 21 - Kill a file

Parameters used:

Filename.

Description:

This allows a caller to delete any file in the download menu which he has uploaded that day. If he uploads a file at 11:45 P.M. and calls back at 12:01 A.M., he will not be able to kill it. Unless he has SYSOP status, he will not be able to delete any other files. (SYSOPs may kill any files in any sections to which they have SYSOP status.)

Application:

This Function is used to delete files from the catalog. Note that it will not delete the file from the disk, just the catalog entry. To kill the file called GRAFPIX.BAS, enter:

K GRAFPIX.BAS

A user may only kill a file he has uploaded on the current calendar day. This prevents someone deciding he no longer wants to use your board, and wants to deprive other users of all the files he has uploaded to you over a period of months or years.

Example(s):

item = K, "Kill A File", -, 6, 21

Function 22 - Add a new user

Parameters used:

User name.

Description:

This Function adds a Member to the permanent user file, giving him the access privileges established with Function 33 (User defaults).

Application:

To add a member, type "A john doe" or "a john doe" and press ENTER. The space after the A is not mandatory. The name will be entered into the system as "John Doe" with the first letter of each name in upper case, the balance in lower case. This is true even if you enter the name as "JOHN DOE."

If you enter "A" without the name, you will be prompted with:

User name:

The new Member's record will be displayed next:

John Doe

A: Location :

B: Phone no. :

C: Time limit: 60

D: Privilege : 10

E: Menu set : 0

F: RD access : 01234567

G: WR access : 01234567

H: UP access : 01234567

I: DN access : 01234567

J: SYSOP :

Enter line to change:

At this point, make any changes necessary. Note that, unlike the name, any address will be saved EXACTLY as entered (new York,ny will be saved as it was typed in). After you have made any changes needed in items A-J, enter a slash ("/"), and the following will be displayed:

1: Terminal : ASCII Text

2: Section : 1

3: Password :

4: Status : Novice

5: Protocol : XMODEM
6: CLS codes : 12 0 0
7: BS codes : 8 0 0
8: Page size : 80 x 24
9: Linefeeds : Yes
0: NULS : 0

You may switch back to the previous set by entering another slash. When you have completed making any changes to the record, press ENTER, and you will see:

John Doe added to user list.

If John Doe is already in the file when you initiate the Add member Function, an error message informs you of the fact. This prevents multiple entries of the same name.

Section Sysops may add Members to the system, or grant access to their sections, but may not alter the record of any Member with a higher privilege level. They also may not raise their own privileges.

Example(s):

item = A, "A: Add A Member", 150-255, 6, 22

Function 23 - Delete a user

Parameters used:

User name.

Description:

This Function removes a user from the user list.

Application:

To remove a user from the permanent user list, enter:

D User Name

Entering the D without the name will cause a prompt to appear asking for it. After you have entered the name, you will be asked:

Are you sure?

If you respond with "Y" the Member will be deleted. If the name is not in the list, you will receive the error message, "User not in file." Check the spelling.

This Function should be limited to SYSOPS. Section SYSOPS will not be permitted to delete any user whose privilege is greater than their own, nor will they be able to delete a user who does not have access to the section(s) for which the Section SYSOP has SYSOP privileges.

Example(s):

item = D, "D: Delete A Member", 100-255, 6, 23

Function 24 - Change a user

Parameters used:

User name.

Description:

This allows you to make changes in a user's record, and displays the information as in Function 22.

Application:

Enter:

C John Doe

You may also enter part of the name, as "John D" or "Joh" and BBS-PC will locate the first name in the list that meets the specification:

John Doe

A: Location : Mudville, NY

B: Phone no. : 212-555-1212

C: Time limit: 60

D: Privilege : 10

E: Menu set : 0

F: RD access : 01234567

G: WR access : 01234567

H: UP access : 01234567

I: DN access : 01234567

J: SYSOP :

Enter line to change:

At this point, make any changes necessary; note that, unlike the name, any address will appear EXACTLY as it was entered (mudville,ny); you may change it now if you desire. After you have made any changes needed in items A-J, enter a slash ("/"), and the following will be displayed:

1: Terminal : ASCII Text

2: Section : 1

3: Password :

4: Status : Novice

5: Protocol : XMODEM

6: CLS codes : 12 0 0

7: BS codes : 8 0 0

8: Page size : 80 x 24

9: Linefeeds : Yes

0: NULS : 0

If the name you entered is not in the list, you will receive the error message, "User not in file." Check the spelling.

Example(s):

item = C, "C: Change Member's Records", 100-255, 6, 24

Function 25 - Purge inactive users

Parameters used:

None.

Description:

This Function allows you to delete users who have not logged onto the system after a given date to prevent carrying users who do not call the system regularly.

Application:

When this Function is activated, you will receive the following prompts:

Purge date: (mm/dd/yy)

Privileges: (low-high)

Are you sure?

The date you enter determines which users are deleted from the system. The privilege range determines which users will be affected by the purge. If you enter a date of 06/01/85 and a privilege range of 1-19, a user with a privilege of 20 will not be deleted no matter how long it has been since his last call. Likewise, a user who has been locked out of the system (privilege 0) will not be purged, thus preventing him from entering again. This Function should be reserved for Wizard SYSOPS. If it is available to section SYSOPS, the purge range will not be higher than that of the SYSOP using it.

Example(s):

item = P, "P: Purge the Deadbeats", 255-255, 6, 25

Function 26 - Purge messages

Parameters used:

None.

Description:

This Function allows you to clear old messages from the message base. Although BBS-PC will scroll the lowest-numbered message off the system (if it is not locked) when the maximum number of messages has been entered, you may use this to keep the message base up to date at any time.

Application:

When this Function is activated, the following prompts appear:

Message number:
Locked messages?
Are you sure?

The message number you enter will be deleted, as well as all lower numbers. If you respond with "N" for locked messages, they will not be affected. This is convenient for keeping certain messages in the system to serve as reminders or notices for New Users, who will see them when they read messages for the first time. Locking a message is one way of making sure that it will not be scrolled off the system, or lost in a purge. If you respond with "Y" for locked messages, they will be affected also.

NOTE: BBS-PC does not require "message packing," and space released by deleted messages is automatically made available for re-use.

Example(s):

item = Z, "Z: Zap the Stale Messages", 255-255, 6, 26

Function 27 - Print user list

Parameters used:

/G, /C, /P, User name.

Description:

Displays the list of all the users in the system, including Guests who have not yet been given Member status. Wizard SYSOPS will see the entire list; Section Sysops will see only those users whose privilege is equal to theirs, or lower. They may not examine the records of users with high privileges.

Application:

The following optional parameters may be used:

L/G = List the Guests only. Only the records of Guests who have registered but have not yet been added as Members will be displayed.

L/C = Continuous display, does not pause after each record is brought to the screen. This allows you to scan the entire list without having to press ENTER to continue after each record is displayed. You may stop it at any time with Control-P. If you wish to make a hardcopy printout of the entire list of users, type L/C, turn the printer on by pressing F10, then press Enter.

L/P = List Users within a range of privileges. You will be prompted for low and high privilege ranges, and see only those users. This is a convenient way of seeing only users who have a special privilege level.

L name = List the users beginning with "name." If you have several users with the name "John" you may start the listing with the first "John." It is not necessary to enter the entire name (you may know only that you want the "John" who is from Kalamazoo, but have forgotten his last name). "L jo" will begin listing the users with the first "Jo," which might be "Jo Anderson," "Joan of Arc," or "Joe Young."

Example(s):

item = U, "L: List Users", -, 6, 27

Function 28 - Reset bulletin flags

Parameters used:

None.

Description:

This Function allows you to reset the flags on any or all user records for reading the bulletins. It will display the following prompts:

Bulletin mask:
Are you sure?

Application:

Once a user sees a bulletin, his record indicates it has been read, and it will not be displayed again. You may reset the flags with this Function. This is usually done when a bulletin has been changed, but you could set the flag daily to force callers to read something that is important, such as an impending change in the hours of operation. Because each section may have its own bulletin, you might change only one of several. For this reason you are prompted for the section mask. Use a plus sign in front of the section flags you want reset:

+1A

resets the flags for Sections 1 and A without affecting any others that are in effect. To reset all flags, enter:

+!

If you had bulletins for sections 1, 2, 3, 4, and 5, and changed all except the bulletin for section 3, you could enter:

+!-3

Example(s):

item = F, "F: Reset Bulletin Flags", 150-255, 6, 28

Function 29 - Update user defaults

Parameters used:

None.

Description:

This Function allows you to change the Time Limit, Privilege, Read, Write, Upload and Download access, Menu set, and Default Section for leaving messages for all users. When activated, this Function displays the following prompts:

Time limit:
Privilege :
Menu set :
RD access :
WR access :
UP access :
DN access :
Section :

Privileges (low-high):
Are you sure?

Application:

This Function will generally be used when you add or delete sections, or change the defaults for all who are currently in the user list. It will not affect users who subsequently are added as Guests or Members unless you change the user defaults (Function 33, User defaults). By setting the privilege range correctly, you are able to update particular classes of users without affecting others. This eliminates the drudgery of updating each individual Member already in the system. Again, note that it affects ONLY those users who are in the permanent user list.

Example(s):

item = G, "G: Global Change User Defaults", 255-255, 6, 29

Function 30 - Define section names

Parameters used:

None.

Description:

This Function allows you to add or remove sections from active use, change the names of the sections, and determine what should be displayed on header during a scan, whether or not there should be a bulletin for that section, allow private messages or not, and set a pseudo-name for uploaded files.

Application:

When activated, this Function displays all sections (0-F), their current settings and names:

Sections:

0:	[DFT—]	Guest Information
1:	[DFTP*]	General Mail
2:	[DFTP*]	IBM-PC/XT/AT/jr
3:	[— — —]	
4:	[— — —]	
5:	[— — —]	
6:	[DFTP*]	Compaq Club
7:	[— — —]	
8:	[— — —]	
9:	[DFT—]	Items for Sale
A:	[DF— —]	Inner-Office Memos
B:	[— — —]	
C:	[DF-P—]	Memos to Director
D:	[— — —]	
E:	[— — —]	
F:	[— — —]	

Section:

Enter the Section number, for example, 1, and you will be prompted with:

Current name : General Mail
New name :

Either enter a new name for the section, or press ENTER, which will retain the current name. The next prompt is:

D,T,F,S,B,*

Enter whichever (or all) of the flags you want set for that section. If set, these indicate what will be displayed with a header while reading messages or scanning headers:

D = Date

T = To

F = From

S = Subject

B = Display Bulletin for this section

* = Use a pseudo-name on files uploaded

After you have entered the flags for the section, press ENTER and you will again be prompted for a section to edit. Press ENTER without a section number, and the prompt for the current menu will be displayed.

To delete a section, enter a space when prompted for the new name.

Note that if you have entered a B for Bulletin, you must have a bulletin on the disk for the section (example, BULL-3.TXT), or an error will occur during execution of the program if a caller with access to that section logs on.

Example(s):

item = N, "N: Section Names", 255-255, 6, 30

Function 31 - Terminal defaults

Parameters used:

None.

Description:

This Function allows you to define terminal types which may be selected by your users, or to edit the defaults once they have been entered.

Application:

Choosing this Function displays the terminal tables, and you may change any of the defaults for any terminal, or add others:

Name	Page	LF	NULS	CLS codes	BS codes	Protocol
0: ASCII Terminal	80 x 24	Y	0	12 0 0	8 0 0	Xmodem
1: RS I, III	64 x 16	Y	0	12 0 0	8 0 0	Text
2: Apple 80-col	80 x 24	Y	0	12 0 0	8 32 8	Xmodem
3: Apple 40-col	40 x 24	Y	0	12 0 0	8 32 8	Xmodem
4: IBM PC/XT/AT/jr	80 x 24	Y	0	12 0 0	8 0 0	Xmodem
5:	80 x 24	Y	0	12 0 0	8 0 0	Xmodem
6:	80 x 25	Y	0	12 0 0	8 0 0	Xmodem
7:	80 x 24	Y	0	12 0 0	8 0 0	Xmodem
8:	80 x 24	Y	0	12 0 0	8 0 0	Xmodem
9:	80 x 24	Y	0	12 0 0	8 0 0	Xmodem

Terminal number:

Select the number and enter any changes you wish. Only those terminals that have been defined will appear as choices when you as SYSOP or a caller asks to change his terminal type. If the caller wanted to edit his terminal type (using Function 4), he would see this:

Terminal type:

0: ASCII Terminal
 1: RS I, III
 2: Apple 80-col
 3: Apple 40-col
 4: IBM PC/XT/AT/jr

To delete an entry, enter a space when prompted for the new name.

Example(s):

item = T, "T: Set Terminal Defaults", 255-255, 6, 31

Function 32 - Modem defaults

Parameters used:

None.

Description:

This Function is used to set your modem(s) for proper operation. This is what you used to install the BBS. Once you have these defaults set, you will not have to change them unless you change modems.

Application:

Selecting this Function displays the current modem settings:

Modem #0

A: Default baud: 0

B: Modem type : 0

C: Setup string:

D: Dial prefix :

E: Dial suffix :

F: 300 result :

G: 1200 result :

H: 2400 result :

Modem #1

I: Default baud: 0

J: Modem type : 0

K: Setup string:

L: Dial prefix :

M: Dial suffix :

N: 300 result :

O: 1200 result :

P: 2400 result :

Any of the current settings may be changed.

Example(s):

item = M, "Change Modem Settings", 255-255, 6, 32

Function 33 - User defaults

Parameters used:

None.

Description:

This Function is used to set or change defaults for Members and Guests.

Application:

Selecting this Function displays the current defaults:

Guests:

A: Time limit: 60
B: Privilege : 5
C: RD access : 0123
D: WR access : 0123
E: UP access : 0123
F: DN access : 0123
G: Section : 0
H: Menu set : 0

Members:

I: Time limit: 120
J: Privilege : 10
K: RD access : 01234567
L: WR access : 01234567
M: UP access : 01234567
N: DN access : 01234567
O: Section : 1
P: Menu set : 0

All users calling after the defaults have been changed will be affected, but not those currently in the user files. To change current users, Function 29 is used.

Example(s):

item = A, "A: Alter Member/Guest Defaults", 255-255, 6, 33

Function 34 - System defaults

Parameters used:

None.

Description:

This Function allows you to change any of the defaults for the configuration of your system.

Application:

When activated, this Function displays the current default settings:

System defaults:

A: Maximum messages: 200
B: Maximum users : 300
C: Maximum uploads : 100
D: Caller log size : 50
E: Caller log range: 0-255
F: Highest menu set: 6
G: Log-in method : 1
H: Upload reward : 0
I: New uploads : Closed
J: Time limit mode : Day
K: Sleep time (min): 5
L: SYSOP password :

Enter line to change:

Any of the current defaults may be changed. If you have created an additional set of menus (using graphics, for example), you must list the highest number in line F. These menu sets are numbered from 0 up, and are included with the name of the menu, as BBS-0.MEN, BBS-1.MEN, etc.

Example(s):

item = S, "Current System Settings", 255-255, 6, 34

Function 35 - Change node defaults

Parameters used:

None.

Description:

This Function allows you to change the manner in which the system operates for a given node. Each node in use can utilize different defaults. You will see the current faults displayed:

Node defaults:

A: File open mode	: IPL
B: Wake-up on call	: No
C: Local node bell	: No
D: Window frame	: On
E: Active COM lines	: 01
F: Polled COM lines	: 01

Application:

The node defaults are set when you initialize the system, but may be changed at any time. Many SYSOPS change items B and C to turn on the screen and the bell while they are close by to monitor activity on the system, then turn them off when they plan to be absent.

A = IPL (Initial Program Load) or LOGIN. Opening all files when the program is loaded initially speeds up the system since no time is consumed in opening and closing files each time one is accessed. The files will remain open until the BBS is properly exited. The LOGIN method opens the files each time a caller logs into the BBS, and closes them when he exits. Because of the time it takes to open the files, we suggest you use the LOGIN method if running BBS-PC on a hard disk, IPL on floppies.

B: Wake-up on call = Yes or No. By selecting the No option, you can prevent screen burn, since the monitor will not turn on each time someone calls. While it will turn off after a few minutes of inactivity, busy systems will find it on for considerable lengths of time, as a new caller will log on before the monitor is turned off. It will stay on for the duration of his call. If it is off when a caller is using the system, it can be turned on by pressing any key.

C: Local node bell = On or off. This has nothing to do with the Chat Function, but provides a way of turning off the bell (or beep) at the local terminal. This can be especially desirable if the BBS is in the same room where you are working and don't wish to be disturbed. Note that it does not turn off the bell at the remote terminal.

D: Window frame = On or off. If you do not care for the frame on the screen, it may be turned off. Because the frame occupies two columns, your effective screen width is 78 columns. If you wish to display the STATS.TXT, for example, 80 columns is necessary.

E and F = Set which COM ports are active, and which one (or both) are checked for incoming calls.

Example(s):

item = N, "N: Change Node settings", 255-255, 6, 35

Functions 36 - List phone directory

Parameters used:

None.

Description:

This Function displays the list of names, phone numbers and communications parameters you have entered:

A: BBS-PC HQ.....	1-305-737-1590	2400:8N1	Line: 1
B: The Notebook.....	655-4713	1200:8N1	Line: 1
C: Big Blue System.....	1-555-1212	300:7E1	Line: 1
D:		0:8N1	Line: 0
E:		0:8N1	Line: 0
F:		0:8N1	Line: 0
G:		0:8N1	Line: 0
H:		0:8N1	Line: 0
I:		0:8N1	Line: 0
J:		0:8N1	Line: 0
K:		0:8N1	Line: 0
L:		0:8N1	Line: 0
M:		0:8N1	Line: 0
N:		0:8N1	Line: 0
O:		0:8N1	Line: 0

Application:

This Function is used from the Terminal Module, and merely shows the current phone numbers, baud rate, paramteres and COM line used.

Example(s):

item = P, "P: List of Remote Phones", 150-255, 6, 36

Function 37 - Change phone listing

Parameters used:

A-O (letter of phone listing).

Description:

This Function allows you to add, delete, or change any of the phones in the list.

Application:

As remote systems change phone numbers, names or parameters, you will want to keep your list up to date. Entering the optional identifying letter of the particular listing you want to change allows you to bypass a prompt asking you for which number to change. Entering "CB" will display the current listing for B and prompt for new information:

```
B: The Notebook.....655-4713 1200:8N1 Line: 0
New name :
New number:
Baud rate :
Parameters:
Modem line:
```

You may retain the current name and number by pressing ENTER at the prompts, but the balance of the information must be entered.

Example(s):

item = C, "C: Change a Listing", 255-255, 6, 37

Function 38 - Dial connect with remote

Parameters used:

A-O (letter of phone listing).

Description:

This Function initiates the dial-out routine for the number you select, setting baud rate, parameters and COM line.

Application:

Unless the optional parameter of the phone designation letter is included with the item to initiate the dial-out routine, you will be prompted for it. If the phone is included, you will see:

XX Dialing XX

You may abort any time until the carrier has been established by pressing Control-P. This will display:

XX No connection XX

Example(s):

item = D, "D: Dial a remote service,", 100-255, 6, 38

Function 39 - Unlisted dial/connect

Parameters used:

None.

Description:

This Function allows you to dial any number that is not in the current list of numbers.

Application:

Although there is room on the list of remote services for 15 numbers, you may have others which are not used frequently enough to be included. When you select this Function, you will be prompted for the number, then the balance of the information:

Number :
Baud rate :
Parameters:
Modem line:

When all the prompts have been responded to, the system will dial the number.

Example(s):

item = U, "U: Dial An Unlisted Number", 150-255, 6, 39

Function 40 - Upload direct

Parameters used:

/Protocol, Filename.

Description:

This Function allows you to send a file directly to the disk from a remote location. It is not limited to files for the download section of the system, but allows you to maintain all disk directories. With the following three Functions, you are able to perform any DOS tasks remotely.

Application:

Many BBS systems are established in a place of business, and the SYSOP is unable to have "hands on" control after hours. You may send any file regardless of type to the BBS and install it directly on the disk. In the case of a new file for the download, it may then be installed in the Download Catalog by using Function 18. You could write new daily bulletins at home and send them to the system; source text for new menus may be sent, then compiled by using Function 43 (DOS gate) and then running the Menu Compiler.

To send a file, enter the name of the file after the item in the menu, as "U MONSTER.COM."

Example(s):

item = U, "U: Upload File to DOS", 150-255, 6, 40

THIS PAGE INTENTIONALLY LEFT BLANK

Function 41 - Download direct

Parameters used:

/Protocol, Filename.

Description:

This Function, like the corresponding Upload Function 40, allows you to retrieve any file on your disk from a remote location. Enter "D filename" to use this Function.

Application:

In conjunction with Functions 40, 42 and 43, this allows you to maintain your system from a remote location. You may copy files remotely that have nothing to do with the BBS and the Download section.

Example(s):

item = D, "Download a File from DOS", 150-255, 6, 41

Function 42 - Kill a file direct

Parameters used:

Filename.

Description:

This Function allows you to delete a file in a DOS directory from a remote location.

Application:

Used in conjunction with the related Functions 40, 41, and 43, This Function permits you to maintain your system totally while at a remote location. You may perform not only BBS maintenance, but any operation, just as though seated at your console. Assuming "K" to be the item option, enter "K filename" to kill a file on the disk.

Example(s):

item = K, "K: Kill a DOS file from remote", 255-255, 6, 42

Function 44 - Search catalog descriptions

Parameters used:

[string].

Description:

Searches for a string in the Description Field of a file in the Download Catalog. This Function finds and displays only those files that have the string in the description.

Application:

If your system carries a large number of files in the download section, a user might be discouraged from reading the entire catalog in a directory. Filenames often tell nothing about what a program will do. If he is interested only in directory utilities, for example, he would be able to search the descriptions in the following manner:

S directory
s directory
sdirect

Note that the option is case independent, and that the space following it is not mandatory. In the third case above, he might find the following files displayed:

MORGAN.EXE	DATE: 03-MAR-85	62387-B	Direct file transfers made easy.
DIRUTIL.COM	DATE: 13-MAR-85	12865-B	Alpha sort of directory.
RBLDDIR.EXE	DATE: 06-JUN-85	44418-B	Rebuild scrambled directories.
BOOKEND.TXT	DATE: 21-JUL-85	3802-T	Directions for binary bookends.

The four files displayed might be four of a hundred in that particular directory, and only two might be of interest to him, but he will have found those two much easier than if he had to search through all the files with the catalog display (Function 15) or browse (Function 16) commands.

Example(s):

item = S, "S: Search for string in description", -, 4, 44

Function 100 - Exit system

Parameters used:

None.

Description:

This Function updates the caller's record, disconnects and returns to the Waiting for Call Menu.

Application:

This is the normal way to terminate a session on a BBS. If a caller simply drops carrier by hanging up, none of his records are updated, and the next time he calls the system will not take into account any changes he made in his records, or the high message he has read. If you have many menus in your system, such as the Main Menu, Download Menu, three or four SIG Menus, a Discussion Menu, Bulletin Menu, etc., you should consider giving your users the option of exiting from each of them. By forcing a caller to retrace his steps through several menus to find the place where he is able to exit, you are inviting him to simply disconnect. Most often used item declarations for this are "E" for "Exit" and "G" for "Goodbye." Do not use "E" for Exit on one menu, "E" for Enter a Message on another, "E" for E-Mail on a third. Consistency and predictability make for happy users.

Example(s):

item = E, "Exit The Notebook", -, 6, 100
item = G, "Goodbye (Terminate Call)", -, 6, 100

Function 101 - Display current menu

Parameters used:

None.

Description:

This Function displays the current menu, and is used primarily by an Expert user to refresh his memory of the options allowed.

Application:

You have total control over menu display. Menus may be displayed if the caller is a Novice, displayed at all times, never displayed, or only if requested. This is determined by the display option in the final field of the Menu Statement. Given the display options:

- 0 = Never display
- 1 = Display for Novice; display for Expert if "?" entered
- 2 = Always display for all users

on the following Menu, the "?" must be included as an item. This is normally a hidden option, using the item display statement [!, ""\]. Most of your Menus should include this Function unless the Menu is not to be displayed:

```
menu = GAMES-O.MEN, "Which Adventure (? for list)", 1
item = ?, !, 6, -, 101
```

Example(s):

```
item = ?, !, 6, -, 101
```

Function 102 - Return specified levels

Parameters used:

None.

Description:

Returns from the Current Menu to another Menu. The final field in the Item Statement (the Argument Field) specifies how many levels to return.

Application:

If you have a number of Menus which lead to other Menus, as in the following Widget Company example, you are able to control the user's movement from one to another:

Main Menu for Widget Company (top level)
Menu for Sales Department (second level)
Menu for Discontinued Items (third level)
Menu for Rookie Salesreps (fourth level)

The argument "1" following the Function number tells the system to return to the previous menu. If the argument were "2" it would "return to the menu you saw before you went to the one that got you to this one." If you begin at the Main Menu, then select each in turn until you are at the Rookie Salesreps Menu, returning one level would display the Discontinued Items Menu; returning two levels the Sales Department menu.

Example(s):

item = Q, " ", -, 6, 102, 1
item = S, "S: Sales Dept. Menu", - 6, 102, 2

Function 103 - Return top level

Parameters used:

None.

Description:

This Function returns control to the Main Menu (BBS-O.MEN). Regardless of the levels of Menus (as in the Widget Company Menus shown in Function 102), this Function ignores previous levels.

Application:

In a system with numerous Menus, it is convenient for the caller to be able to find his way back to the Main Menu which should serve as a master control center. From the Main Menu, he could move to the Download, Bulletins, E-Mail, Utilities, Discussions, or any other Menu, any of which can have sub-menus. This provides a fast way to get back to the "control center."

Example(s):

item = M, "M: Flying Dutchman's Master Menu", -, 6, 103

Function 104 - Call another menu

Parameters used:

None.

Description:

This Function changes control to a different menu, with the ability to return from the Menu called, functioning much like a GOSUB with a RETURN.

Application:

If your menus were set up as follows:

Main Menu for Widget Company (top level)
Menu for Sales Department (second level)
Menu for Discontinued Items (third level)
Menu for Rookie Salesreps (fourth level)

and from the Main Menu you called the Menu for Sales Dept., then called the Menu for Rookie Salesreps, from the Rookie Menu You would be able to return to the Main Menu with Function 103, or to the Sales Department Menu using Function 102 with the argument "2" to return two levels.

Example(s):

item = D, "D: Discontinued Items", -, 6, 104

Function 105 - Goto another menu

Parameters used:

None.

Description:

This Function changes control to another Menu. Unlike Function 104 (Call another menu) you cannot return to a previous level. You must either "goto" or "call" another menu.

Application:

This will probably be the most frequently used method of transferring control of BBS-PC from one Menu to another. Keep in mind that the system need to retain information on Menus called (Function 104) in order to return to specific Menus (Function 103). Careless use of called Menus without returns could result in a stack overflow. Simply moving to a different Menu does not present this problem.

Example(s):

item = F, "Open Forum Discussions", 10-, 6, 105

Function 106 - CLS, type a file

Parameters used:

None.

Description:

This function clears the screen and displays a text file. The name of the file must always appear as an argument in the final field of the Item Statement: (106, SAMPLE.TXT) in the menu item.

Application:

There are many places within your BBS where you would want to display a text file. The Information and Help files for the BBS and the Download section are but two of them. The more complex your system, the more helps you will want to offer your users. If you have a Discussion section, or an E-Mail section, for example, you need to inform the user what the rules for use of those sections are, and additionally tell him how to use them. A system that has a large number of private pilots might want to post frequently-updated weather reports, and make this available from the Main Menu. Rather than do this in a bulletin such as BULL-O.TXT, you could have it as a Menu selection, preserving the bulletins for more general or newsworthy items.

Many systems feature such things as movie reviews to keep their local users informed of what movies are worth watching (at least in the reviewer's personal and/or biased opinion). Local events such as rock concerts in the area can be noted in a text file, and made available with this Function.

Example(s):

item = W, "W: Weather conditions", -, 6, 106, WEATHER.TXT
item = M, "M: Movies - The Guru's Choice", -, 6, 106, REVIEWS.TXT
item = V, "V: Best Values at Local Supermarkets", - 6, 106, SHOPPING.TXT

Function 107 - Type file

Parameters used:

None.

Description:

This function displays a text file without first clearing the screen. The name of the file must always appear as an argument in the final field of the Item Statement: (107, SAMPLE.TXT) in the menu item.

Application:

There are many places within your BBS where you would want to display a text file. The Information and Help files for the BBS and the Download section are but two of them. The more complex your system, the more helps you will want to offer your users. If you have a Discussion section, or an E-Mail section, for example, you need to inform the user what the rules for use of those sections are, and additionally tell him how to use them. A system that has a large number of private pilots might want to post frequently-updated weather reports, and make this available from the Main Menu. Rather than do this in a bulletin such as BULL-O.TXT, you could have it as a Menu selection, preserving the bulletins for more general or newsworthy items.

Many systems feature such things as movie reviews to keep their local users informed of what movies are worth watching (at least in the reviewer's personal and/or biased opinion). Local events such as rock concerts in the area can be noted in a text file, and made available with this Function.

Example(s):

item = W, "W: Weather conditions", -, 6, 107, WEATHER.TXT
item = M, "M: Movies - The Guru's Choice", -, 6, 107, REVIEWS.TXT
item = V, "V: Best Values at Local Supermarkets", - 6, 107, SHOPPING.TXT

Function 108 - CLS, type a file without "- more (Y/N/NS)"

Parameters used:

None.

Description:

This Function works like 106, but will not pause and display the prompt at the end of a page. This ignores page length for continuous display of a text file.

Application:

There are many places within your BBS where you would want to display a text file. The Information and Help files for the BBS and the Download section are but two of them. The more complex your system, the more helps you will want to offer your users. If you have a Discussion section, or an E-Mail section, for example, you need to inform the user what the rules for use of those sections are, and additionally tell him how to use them. A system that has a large number of private pilots might want to post frequently-updated weather reports, and make this available from the Main Menu. Rather than do this in a bulletin such as BULL-0.TXT, you could have it as a Menu selection, preserving the bulletins for more general or newsworthy items.

Many systems feature such things as movie reviews to keep their local users informed of what movies are worth watching (at least in the reviewer's personal and/or biased opinion). Local events such as rock concerts in the area can be noted in a text file, and made available with this Function.

Example(s):

item = M, "M: Movie Reviews", -, 6, 108, MOVIES.TXT

Function 109 - type a file without "- more (Y/N/NS) -"

Parameters used:

None.

Description:

This is like Function 107, but does not clear the screen before the continuous display of the file, and does not pause at the end of the page and display the "- more (Y/N/NS)" prompt.

There are many places within your BBS where you would want to display a text file. The Information and Help files for the BBS and the Download section are but two of them. The more complex your system, the more helps you will want to offer your users. If you have a Discussion section, or an E-Mail section, for example, you need to inform the user what the rules for use of those sections are, and additionally tell him how to use them. A system that has a large number of private pilots might want to post frequently-updated weather reports, and make this available from the Main Menu. Rather than do this in a bulletin such as BULL-0.TXT, you could have it as a Menu selection, preserving the bulletins for more general or newsworthy items.

Many systems feature such things as movie reviews to keep their local users informed of what movies are worth watching (at least in the reviewer's personal and/or biased opinion). Local events such as rock concerts in the area can be noted in a text file, and made available with this Function.

Example(s):

item = M, "M: Movie Reviews", - 6, 109, MOVIES.TXT
item = W, "W: Local Weather Outlook", -, 6, WEATHER.TXT

Function 110 - Change menu set

Parameters used:

None.

Description:

Permits a user to switch from the default menu set to an alternate set if multiple menu sets are available. Also permits him to switch back to the default, or to any other available set.

Application:

The default menu set is 0, which should display properly on almost any terminal, but you are able to design menus utilizing the graphics capabilities of any specific computer. Each of the menus needs to be rewritten and numbered with the new set. For example, you could have:

BBS-0.MEN
BBS-1.MEN
BBS-2.MEN
BBS-6.MEN

on your disk, and the caller will see the alternate menu sets automatically when he registers his terminal type (Function 3, Register/Add to membership file). He can switch to a different set of menus with this Function, or with Function 4 (Change/examine user record).

When a user selects this option, he will see a list of the different types of Menus available:

0: Standard ASCII
1: TRS-80 I, III
2: IBM
6: Commodore 64

When he selects the correct display for his terminal, he will then see the graphics display designed for his computer. If he has an IBM terminal, he would automatically have Menu Set #2, but if he also called from a Commodore C-64 terminal, he would want to select Menu Set #6. (There are adventuresome types who want to experiment with everything. If an Apple user attempted to use Menu sets 1, 2, or 6, he would get a screen full of garbage. This does no harm either to the BBS or to the remote caller, but it will provide him with something different to look at.)

Alternate Menu sets do not have to employ graphics, although this is one of the most-often used reasons for creating them. If you have a large number of Apple and Commodore users, you might want one set of Menus that display a maximum width of 39 characters (the 40th column is reserved on those systems for the Carriage Return). Your IBM and Kaypro users could have Menus that

employ the full width of an 80-column screen, and Radio Shack users Menus a maximum of 64 columns wide. See additional remarks in the Menu Compiler Section of this manual.

Example(s):

item = M, "M: Change to a Different Menu Display, -, 6, 110

Part IV

BBS-PC! Support Utility Programs

Introduction

Supplied with BBS-PC! 4.0 are a number of utility programs designed to make the operation of your bulletin board simpler, as well as to allow you to recover most of the damages caused by scrambled information on your disks. Some of the programs you will need to use only once; others will be used frequently as you make changes in your Menus, compile up-to-date statistics, and maintain the system. These files are:

BBSINIT.COM	Initializes or reinitializes the system.
BBSMENU.COM	Compiles Menus from a source text file.
BBSINFO.COM	Creates a text file showing current statistics.
BBSFIX.COM	Repairs files, indexes, counters, etc.
BBSFILE.COM	Writes descriptions to download files.
CHKFILE.COM	Provides useful SYSOP utilities for mass file maintenance.

BBSINIT and BBSMENU are covered in some detail in Sections I (Setup and Installation) and Section II (Customizing Your BBS with the Menu Compiler), but are included in this section for reference.

BBSINIT.COM

This program is used to create the data files necessary for your BBS. From DOS type:

BBSINIT <ENTER>

You can also specify an optional switch that indicates you wish to only update the pathnames for the upload/download files. BBS-PC! can look in as many as 16 different drives or directories for its upload/download files. Those pathnames are set and modified with this utility. To modify the current settings, type:

BBSINIT -U <ENTER>

The "-U" switch will cause BBSINIT to display the current upload/download pathnames and move directly to the prompt requesting the new pathnames.

The BBSINIT program will display the following:

BBSINIT - BBS-PC! Initialization program - 4.0
Copyright (c) 1985, Micro-Systems Software Inc.

Generate parameter file?

Responding with a "Y" will cause several prompts to appear. The first four concern where to place the BBS data files NOT in the default drive or directory; the fifth offers you the option of updating ONLY the location of the BBS files and not creating them. We'll cover that in a moment. The information about where to locate the various divisions of the data files is stored in its own data file. This is the BBS.P file (the "parameter file"). This file is very important and must remain the default directory along with the BBS-PC! file, as it tells the program where to find its data files. A response of "Y" here tells BBSINIT that you do want to create this file. Type: Y <ENTER>. You'll see the first prompt.

Drive/path for message files:

This path specifies the location of the message base data files. These files begin with the letters "MSG".

Drive/path for user files :

This path specifies the location of the permanent user log data files. These files begin with the letters "USER".

Drive/path for U/D files :

This path specifies the location of the upload/download indexes and description files. These files start with the letters "UD". These are NOT the actual upload/download files themselves. Simply the catalog.

Drive/path for log files :

This path specifies the location of the caller log files. These files start with the letters "CALL". They record the log of recent callers.

Generate main BBS files?

This question is asking if you want to actually CREATE the data files or if you are simply telling the BBS where to find them again. Since we are setting up for the first time, type: Y <ENTER>. BBSINIT will then create all the system data files. The next prompt is:

Enter up to 16 paths for U/D files
Drive/path #1:

This next series of questions is asking you what drive(s) or directory(s) you will use to hold the actual upload/download files themselves. These are not to be confused with the file catalog indexes and descriptions, which are stored in the location specified at the "U/D files" prompt above. Whenever a caller requests to download a file listed in the catalog, BBS-PC! will look

in all 16 of these paths before informing the caller that the file is not on-line. In addition, all new uploads will be stored in the first of these paths. You may want to use a separate directory name for new uploads only.

Do not worry about adding paths or changing existing ones. This can be done later by using BBSINIT with the "-U" switch. After you press <ENTER> you will be asked for additional Drive/paths.

Pressing <ENTER> alone at one of the prompts will terminate the upload/download path designation, and bring you to the final prompt:

Initialize BBS node file?

Responding with a "Y" will display:

Which node (1-99)?

Because this is the primary node, answer with a "1". (The great majority of systems will have only one node, not being part of a network.) Additional nodes in a multi-user system will carry numbers designating their respective locations in the network. Each node can be configured separately with respect to COM ports, modems, and display characteristics.

As soon as all questions are answered, BBSINIT will create all of the needed files using the paths that you have specified. When the DOS command prompt re-appears, your BBS-PC! system will have been created. If for any reason you decide that you would like to "start over", you must first delete all of the files with .DAT pattern in the filename.

BBSMENU.COM

Menus must be compiled before they can be used with BBS-PC!. This is done with BBSMENU.COM. When you write your Menus, we suggest you save them with an extension of .TXT or .DOC, for example, MENUS.DOC. You can store all your Menus in one source text file, or have a separate file for each. (We'll show advantages of both later.) To compile them, simply enter this at the DOS prompt:

BBSMENU MENUS.DOC

or

bbsmenu menus.doc

The source text in MENUS.DOC will then be compiled, and create one or more Menus with the extension .MEN and write them to the disk. You need a minimum of four Menus, which are:

BBS-O.MEN	(Main system Menu)
UD-O.MEN	(Upload/Download Menu)
TM-O.MEN	(Terminal Menu)
SM-O.MEN	(Maintenance Menu)

While these are the minimum required to operate BBS-PC!, you will undoubtedly have many more, such as:

BULL-O.MEN	(Bulletins and Help file Menu)
UTIL-O.MEN	(Utility Menu)
SYSOCM-O.MEN	(Comments to SYSOP Menu)
KAYPRO-O.MEN	(Kaypro SIG Menu)
EXIT-O.MEN	(Exit choices Menu)

Some of these can be very short, such as the Exit Menu, which simply asks, "Do you want to leave us a comment (Y/N)?" as the caller logs off. If you have set aside a section for an IBM-PC User's Group, you could even have a Menu such as PCUG-O.MEN for those users, which is totally different from the Menus seen by other users, and which is as full as your main Menu.

The source text for your Menus contains a list of several things:

- A "Menu" Statement
- One or more "Title" Statements
- One or more "Item" Statements
- One or more "Item" Statements
- An "Endmenu" Statement.

The Menu MUST begin with a Menu Statement and end with an Endmenu Statement. There may be one, none or several of each of the other statements. The following Menu contains one of each except a title:

```

menu = SYSCOM-0.MEN, "Is this private to the SYSOP (Y/N)?", 0
iret = Y, "\, -, 3:C, 10
item = N, "\, -, 6, 102, 1
endmenu

```

The Menu Statement

Let's look at the Menu Statement first. It consists of three fields, each of which is separated from the next by a comma. The space following the comma in these examples is optional, but using the space makes the text easier to read.

```
[menu = SYSCOM-0.MEN], "Is this private to the SYSOP (Y/N)?", 0
```

```
menu = SYSCOM-0.MEN
```

This is the "declaration field," and indicates the beginning of a Menu called "SYSCOM-0.MEN."

```
menu = SYSCOM-0.MEN, ["Is this private to the SYSOP (Y/N)?"], 0
```

```
"Is this private ...."
```

The "prompt" field, which contains the prompt to be displayed for the Menu. The prompt MUST be enclosed within quotes.

```
menu = SYSCOM-0.MEN, "Is this private to the SYSOP (Y/N)?", [0]
```

```
0
```

The "display" option, which states how the Menu will be displayed, if at all.

All Menu Statements must contain those three fields. The third field, the Display Option, defines how the Menu will be displayed:

0 = Never display

1 = Display Menu automatically for Novice.

2 = Display Menu before command prompt for all users.

Display Option 0 allows you to have a prompt with no Menu displayed, as in the example above. Because the only choices are Y and N, there is no need to display anything except the prompt.

Display Option 1 will display a Menu for Novices, but not for Expert Users, who see only the prompt. This is the most frequently-used option for Menus.

Display Option 2 will always display the full Menu regardless of the status of the user. This is generally used on Menus that are seldom used, or in cases where you want to display a brief message prior to getting the user's input.

The Title Statement

A Title is a line that is displayed without defining an item. We could have written the EXIT-O.MEN Menu as follows:

```
menu = SYSCOM-O.MEN, "Is this private to the SYSOP?", 2
title = "Please leave private mail to Members"
title = "in the E-Mail Section."
title = ""
item = N, "N: No.", -, 6, 102, 1
iret = Y, "Y: Yes.", -, 3:C, 10
endmenu
```

When this Menu is displayed, it will look like this:

```
Please leave private mail to Members
in the E-Mail Section.
```

```
Is this private to the SYSOP?
```

The display option 2 states it will always be displayed, for all users regardless of status. Note that the third Title statement is a null (""); this will display a blank line in the display.

Title Statements may be followed by a privilege range, and this will be discussed below in the "Privilege Field" portion of the Item Statements.

Titles are most commonly used as headers on menus, although they are not confined to that. They may be used to display text anywhere within a menu, or blank lines. Later we'll show you another way of writing a Menu with all display statements created by Titles, and all Item statements hidden from view.

The Item Statement

The Item Statement is the core of the Menu definition. Each Item Statement consists of five or six fields, each of which MUST be separated from the next with a comma; a space following the comma is optional:

```
item = E, "E: Exit System", 10-, 6, 100
item = E, "E: Exit System", -9, 6, 105, EXIT-O.MEN
```

The first field is the Item Definition field, and contains the single letter or other character that will call the Function:

[item = E], "E: Exit System", 10-, 6, 100

This defines the letter E as the "Key" which will activate the Function in the Function field.

The next field is the Display Description field:

item = E, ["E: Exit System"], 10-, 6, 100

This is what determines what is displayed on the Menu, and will appear as follows:

E: Exit System

Following the display field is the privilege level field:

item = E, "E: Exit System", [10-], 6, 100

item = E, "E: Exit System", [-9], 6, 105, EXIT-O.MEN

In the first example, only a Member (with a privilege level of 10 or greater) will activate Function 100 (Exit system); Guests and New Users (privilege levels below 10) will go to another Menu prior to exiting. This Menu (EXIT-O.MEN) is where he could be asked if he wants to leave a comment.

The fourth field is the Access Field. There are eight levels of access:

- 0 = Requires specified section access
- 1 = Requires specified access with mask
- 2 = Requires RD access with mask
- 3 = Requires WR access with mask
- 4 = Requires DN access with mask
- 5 = Requires UP access with mask
- 6 = No section access required
- 7 = Requires SYSOP password, if set.

Neither of the Exit items above requires particular section access, and is available to anyone within the proper privilege levels.

Before we show examples of other Access levels, let's continue with the next field, the Function Field:

item = E, "E: Exit System", 10-, 6, [100]

item = E, "E: Exit System", -9, 6, [105], EXIT-O.MEN

In the first example, choosing the option E will activate Function 100 (Exit system). The second example activates Function 105 (Goto another menu), and requires a final, or sixth field, called the Argument Field:

item = E, "E: Exit System", -9, 6, 105, [EXIT-O.MEN]

The Argument Field contains the name of a program, data or text file or may contain a numeric constant. Only the following Functions admit an Argument:

- ```

102 - Return specified levels
104 - Call another menu
105 - Goto another menu
106 - CLS, type a file
107 - Type a file
108 - CLS, type a file without "- more (Y/N/NS)"
109 - Type a file without "- more (Y/N/NS)"
110 - Change menu set
111 - Execute external program

```

These Functions, and only these Functions, will be followed by an Argument. The Argument, whether a Menu or Text File, is never enclosed within quotes.

## The Iret Statement

An Iret Statement consists of the same fields as an Item Statement. In the SYSCOM-0.MEN above, it appears this way:

iret = Y. "n\, -, 3:C, 10

The only difference between an Item and an Iret, is that the Iret Statement means, "Perform the Function indicated, then RETURN to the Menu from which this Menu was called." Because only the prompt is seen offering a choice of Yes or No in this example, there is no need to display anything else. If this were to display anything, the Display Description field would be filled in within the quotes.

## The Privilege Field

The third field in an Item Statement or Title Statement is the Privilege Field. This determines not only who can select the option, but also who can even see it displayed. Take the following for example:

```
item = E, "E: Exit System", 10-, 6, 100
title = " ", -9
title = "Guests are asked to leave a comment", -9
title = "when you exit. We'd like to know how", -9
title = "you like this system.", -9
title = " ", -9
item = E, "E: Exit System", -9, 6, 105, EXIT-0.MEN
```

Of the eight lines, a Member (Privilege 10 or greater) will see only the first line displayed:

E: Exit System

New Users and Guests (anyone whose Privilege is less than 10) will see:

```

+-----+
| Guests are asked to leave a comment |
| when you exit. We'd like to know how |
| you like this system. |
+-----+

```

E: Exit System

In this manner, Guests and Members see different parts of the Menu, and with multiple Item Statements as shown here, selecting the option E will bring different Functions into play. Your only limitation is that you may have a total of no more than 50 Item, Iret and Title Statements in any one Menu. Seldom will this many be necessary, for your Main Menu could direct Members and Guests to different Menus for the duration of their call on the board.

### The Access Field

The fourth field in an Item or Iret Statement is the Access Field, and determines what type of access is necessary to activate the Function. The access levels are:

- 0 = Requires specified section access
- 1 = Requires specified access with mask
- 2 = Requires RD access with mask
- 3 = Requires WR access with mask
- 4 = Requires DN access with mask
- 5 = Requires UP access with mask
- 6 = No section access required
- 7 = Requires SYSOP password, if set.

The first six Access Levels require setting the specific mask as well as the Section Number. The Section Number is appended to the Access Level Number with a colon:

```

item = R, "R: Read Messages", -, [2:-BC], 11
item = I, "I: IBM SIG", 25-, [2:B], 104, PCUG-O.MEN
item = K, "K: Kaypro SIG", 25-, [2:C], 104, KUG-O.MEN

```

Note that in the first example, Sections B and C are removed (with the "-"), allowing all other sections to be read (B and C are "masked out"). In the next two examples, which are seen only by users with a privilege level of 25 or higher, Sections B and C are "masked in" and the other sections are masked out. If a user does not have Read Access to these sections, he will not even see the option displayed.

In the Exit examples shown, no specific section access is required, as shown by the "6" in the field:

```
item = E, "E: Exit System", 10-, [6], 100
item = E, "E: Exit System", -9, [6], 105, EXIT-O.MEN
```

Often you will want to limit what a user may do by restricting the access level. For example, if you set section B aside for a SIG such as the PC Users Group for members only, you could prevent other users from seeing any messages in that section in the following manner:

```
item = R, "R: Read Messages", -, [2:-B], 11
```

The access level of 2 indicates the user must have Read Access to see any messages, and that section B will be masked from view. If Section B were an IBM SIG and Section C a Kaypro SIG, to prevent users who are not members of those groups from seeing any messages in those sections, use the following:

```
item = R, "R: Read Messages", -, [2:-BC], 11
```

Note that even with Read access to these sections, a user will not be able to see the messages. He must choose an option from the Menu which sets his mask to those sections:

```
item = I, "I: IBM SIG", 10-, [2:B], 104, PCUG-O.MEN
item = K, "K: Kaypro SIG", 10-, [2:C], 104, KUG-O.MEN
```

Some Functions can be protected from use by anyone who does not know the SYSOP Password, which you set when you set up your BBS-PC!. Access Level 7 is used for this:

```
item = A, "A: Apply for Membership", 1-4, [7], 3:10
```

Only New users will see this, and if they select this option from the Menu, they will be prompted with:

SYSOP Password:

An invalid password will abort the Function and return to the Menu.

One additional item worth mentioning is in several instances BBS-PC! may be queried to display section names as in Functions 0, 1 and a few others which may not be as obvious. The sections/access privileges shown by BBS-PC! depend on the access type used as part of the function. Using access type 6 for example will force a display of all sections since no specific section access is required. You may find it more desirable to use access type 0 for these functions.



## Control of Display

Further control of how something is displayed is available following the Item Definition field:

|                |                                                                                   |
|----------------|-----------------------------------------------------------------------------------|
| "Description", | Displays text within the quotes.                                                  |
| !,             | Hides the Item from view.                                                         |
| ""             | Produces a blank line.                                                            |
| ""\,           | Used when the Menu is made up of Titles, with Item definitions listed separately. |

Compare the following two ways of writing a Menu:

```

***** Notebook Main Menu - 08/21/85 *****
menu = BBS-O.MEN, "Command (? for menu): ", 1
item = ?, !, -, 6, 101
title = "-----The Notebook's Main Menu-----"
item = B, " | B: Bulletins/Help | C: Chat w/Sysop | "-, 6, 104, BULL-O.MEN
item = E, " | E: Exit Notebook | L: Leave Message | ", 10-, 6, 100
item = E, " | E: Exit Notebook | L: Leave Message | ", -9, 6, 105, EXIT-O.MEN
item = R, " | R: Read Messages | N: Section Names | ", -, 2:-3CE, 11
item = X, " | X: Menus on/off | #: Note to Sysop | ", -, 6, 8
title = "-----"
item = C, ""\, -, 6, 7
item = L, ""\, -, 3, 9
item = N, ""\, -, 0, 1
item = #, ""\, -, 6, 104, SYSCOM-O.MEN
endmenu

```

\*\*\*\*\* Notebook Main Menu - 08/21/85 \*\*\*\*\*

```
menu = BBS-O.MEN, "Command (? for menu): ", 1
title = "-----The Notebook's Main Menu-----"
title = " B: Bulletins/Help C: Chat w/Sysop "
title = " E: Exit Notebook L: Leave Message "
title = " R: Read Messages N: Section Names "
title = " X: Menus on/off #: Note to Sysop "
title = "-----"
```

```
item = ?, 1, -, 6, 101
item = B, "", -, 6, 104, BULL-O.MEN
item = E, "", 10-, 6, 100
item = E, "", -9, 6, 105, EXIT-O.MEN
item = R, "", -, 2:-3CE, 11
item = X, "", -, 6, 8
```

```
item = C, "", -, 6, 7
item = L, "", -, 3, 9
item = N, "", -, 0, 1
item = #, "", -, 6, 104, SYSCOM-O.MEN
endmenu
```

In the second example, everything displayed is written with a Title Statement, and the Item Definitions follow. There is much to be said for the second method, for it allows you a visual check on the appearance of what is displayed. In the first example (part of an actual Menu used by the author of this manual when he initially set up his system), two lines were out of alignment with the others:

```
+-----The Notebook's Main Menu-----+
| B: Bulletins/Help C: Chat w/Sysop |
| E: Exit Notebook L: Leave Message |
| R: Read Messages N: Section Names |
| X: Menus on/off #: Note to Sysop |
+-----+
```

Command (? for menu):

It is frustrating to write and compile a Menu only to find one or two things do not look right. Writing Menus as Titles followed by Item Definitions eliminates this problem.

The addition of the backslash to a null display statement ("") suppresses the carriage return at the end of the line. If two Items in the above Menu lacked the backslash, the Menu would appear in this fashion:

| The Notebook's Main Menu |                  |
|--------------------------|------------------|
| B: Bulletins/Help        | C: Chat w/Sysop  |
| E: Exit Notebook         | L: Leave Message |
| R: Read Messages         | N: Section Names |
| X: Menus on/off          | #: Note to Sysop |

Command (? for menu):

Note that the question mark is not displayed, because it is hidden by the "!" in the Display Description field. This allows you to have any number of concealed items which will never be seen by a user.

Because the Menu Display Option (the "1" in the third field of the Menu Statement) instructs BBS-PC! to display the Menu only for Novices, an Expert will see only the prompt:

Command (? for menu):

If an Expert presses Enter, he will see:

Options: BCELRNX#

Entering a "?" will display the full Menu. If a Novice presses Enter, the Menu will be displayed again.

#### The Endmenu Statement

The last line in the source text for a Menu is the Endmenu Statement:

endmenu

There are no parameters. Without an Endmenu Statement, the compiler would have no way to know it had completed all lines of the Menu source text.

#### The Comment Statement

If you care to document your source text by including lines that will not be acted upon by the compiler, simply begin them with a period:

.\*\*\*\*\* Notebook Main Menu - 08/21/85 \*\*\*\*\*

When the Menu Compiler encounters a period as the first character of a line, it ignores the line and proceeds to the next. In the second example of The Notebook Menu, three lines consist only of a period, followed by a carriage return. This simply separates portions of the Menu visually, making it easier for you to group the Item Statements according to their location on the Menu.

If you create Menus that are as wide as your screen, with three or four Items on one line, doing it with Titles and grouping the Items below is perhaps the best way. Consider the author's first SM-0.MEN (Sysop Maintenance Menu):

#### The Director's Control Room

| The Membership          | The System               | The Tunnels         |
|-------------------------|--------------------------|---------------------|
| A: Add a New Member     | B: Display Statistics    | 1: Back Door to BBS |
| C: Change Member Record | F: Flush Stale Messages  | 2: The Phone Booth  |
| D: Delete One Member    | M: Check Modem Settings  | 3: Slip Over to DOS |
| L: List All Members     | N: Change Section Names  | 4: Go Wait for Call |
| L/G: List the Guests    | R: Reset Bulletin Flags  | 5: Lending Library  |
| P: Purge the Deadbeats  | T: Check System Defaults | 6: Open Forum       |
| S: Check User Defaults  | V: Change Mode Settings  |                     |
| U: Update All Users     | W: Change Menu Sets      |                     |
| Y: Caller Log           | Z: Edit Terminal Setups  |                     |

In this Menu, at least 12 of the Item Statements need to be placed after the Display because only one Item may appear on any line. Placing all Item definitions after the Titles and grouping them by Membership, System, and Tunnels makes it simple to write, and prevents overlooking one. (It is easy to do when you have more than one Function on a line.)

#### Compiling the Menus

When you have created your source text for your Menus, compiling them is as simple as stated early in this section. Be certain you begin with a Menu Statement and finish with an Endmenu Statement. From the DOS prompt, enter

```
bbsmenu menus.doc
```

The compiler will then compile the source text one line at a time, displaying each line on the screen as it does so. Any errors in the source text will cause it to stop compiling and display an error message. (All error messages you may encounter in running BBS-PC! or the Menu Compiler are listed in Appendix \*\*\*) The most common errors are:

Syntax errors. Generally they are one of two types (although not limited to two): Either you have neglected to place a comma after each field, or you have used a period instead of a comma. These two account for about 75% of the errors encountered.

Another common error is neglecting to place Display Statements within quotes. The opening quotation mark is usually there, but the closing quote will be missing. Failure to have quotation marks at both beginning and ending of the Display Statement is guaranteed to create havoc during the compiling of a source text.

Another error is attempting to put too many Items and Titles in a Menu. The BBS-PC! Menu Compiler is a limited, special purpose compiler. You are limited to 40 Items and Titles. More than this will cause the compiler to quit with an error message.

When the compiler has finished, it writes the Menu(s) to disk with the .MEN extension. Be sure you run BBS-PC! and check each option on every Menu to make certain they operate as you intended. Incorrect Function Numbers, Access Levels, Display Options, Privilege Levels, etc., will not be corrected by the Menu Compiler. A simple typographical error in entering a Function Number can ruin the best of days if not tested before going on line. Imagine how you would feel if you typed "23" instead of "13" and a caller wanted to Scan the message headers. He would see "User name:" and likely would type in his own name. What he has done instead of scanning the headers was delete himself from the permanent user log!

### Creating Graphic Menu Sets

You can use graphics in your menus not only for your own computer, but for any computer for which you have the ASCII special character codes. With a good text editor this Menu

```
+-- The Hidden Valley Kaypro Club Message Center -----+
|
| 0 = Read Public Bulletins 5 = Kaypro Club Members Only |
| 1 = Post a Public Bulletin 6 = Current Weather Forecast |
| 2 = File Transfer System 7 = Movie Reviews by the Guru |
| 3 = Hidden Valley Helps 8 = Hidden Valley Utilities |
| 4 = Information on Membership 9 = Sign Off for Today |
|
+----- Contact Ed Jones at 555-1212 for more information --+
```

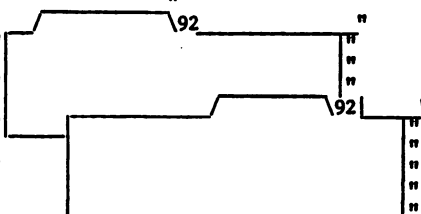
can be converted easily to ASCII character graphics. Some text editors use the the search and replace method (searching for the "|" and replacing it with the proper decimal code) and others allow you to enter the graphics directly. Even ANSI escape sequences for color menus are allowed. The only restriction is that no single line can exceed the maximum allowed length.

If your text editor is not capable of replacing all of them, you can write the source text code to have the compiler insert them, by preceding the code with a backslash ("\"). In the file card menus shown below, the lines with backslashes were written like this:

```

title = "
title = "
title = "
title = "
title = "
title = "
title = "
title = "
title = "
title = "

```



The backslash is ASC(92), and when the Menu Compiler encounters the typed backslash it looks for an ASCII code to follow; when it finds the "92" it prints the backslash.

In designing graphic menus, consider the majority of your users. While most of the author's users are pleased with the appearance of the filecards created with vertical bars, slashes, backslashes and underlines, his Commodore-64 users have a problem, for the underlines appear on their screens as "<-" (left arrows). Except for the login text (which is also a file card), there is an alternate set of Menus for the C-64 users, which simply substitutes hyphens for the underlines.

Unless you are familiar with all the subsets of ASCII special characters as implemented on different systems, the easiest and most considerate way to design alternate Menu sets is to ask one of your users who has a different machine to help you with the various codes.

## BBSINFO.COM

In order for you to know exactly what has been going on with your BBS, the utility program BBSINFO searches the data files and creates a file called STATS.TXT. To run this program, enter:

## BBSINFO

You will be prompted with:

Hours/day of operation:  
Previous period filename:  
New period filename:

The first prompt is asking you for the average number of hours that your system is online each day. If your system is running 24 hours a day, you may press <ENTER> with no other input. If you have a 24 hour system, and normally have it off-line for an hour or so while you perform maintenance, or if you use the computer for other tasks, enter a figure that reflects the actual number of hours the system is available to callers. Having this number input correctly will cause the "Average Percentage of Hourly Usage" graph to appear correctly later.

BBS-PC! automatically ages the usage and caller statistics based upon the current caller log. If you want a larger period in your statistics file, all you have to do is increase the size of the caller log. However, for the file transfer statistics, the aging must be done manually. This is handled via the "previous" and "new" periods.

Each period can be whatever length of time you desire. To see the file activity since a particular time, input the name of the period file. Then the file statistics in STATS.TXT will reflect only what has CHANGED since the period file was created. The "new period" prompt is how you freeze a point in time (taking a "picture" of the upload/download statistics, if you will). The file created there becomes a "previous period" file for future reference.

The first time you run BBSINFO, obviously, simply ENTER for the "Previous period filename". If you want to compare the file upload/download statistics of today against some future time, enter a filename for the new period. The "New period filename" should reflect the current date so you may use it for comparisons of statistics on succeeding runs. The name may be anything, such as AUG-24, or INFO8-24, or even INFO-TUE. No extension is necessary. Both of the prompts are optional. If omitted, file activity since the start of the system will be reported.

The file created by this program is STATS.TXT, and it may be displayed by entering:

type STATS.TXT

This text file could also be listed with Ed or displayed via a "type file" function on one of your menus. Let's look at some of the components of these charts and graphs. (It may be helpful for you to run this program once and have the text file on the screen at this point.)

The "Average Percentage of Hourly Usage" shows the percentage of time used for each individual hourly period over the course of the time charted. (In this case, it was 7 days. This is controlled by the size of the caller log.) Each hour is considered independent of all other hours. (For example, if the percentage for 11 p.m. shows 100%, this would indicate that during the 11 p.m. hourly period, for the 7 days charted, the system was in use ALL the time with no interruption. It is not the percentage of total time used.

Following that graph are the "Communication Statistics". This lets you see how many calls people are making at the different baud rates and serial settings supported by the board and what sort of activity has been going on. The "Percentage of utilization" will be more accurate if the "Hours of operation" prompt was answered correctly.

The next graph, "Average Percentage of Section Usage", shows you how your board's sections are being used, message versus files. After your system has been in operation for some time, you might find one or more sections that receive very little use. This could be an indication that it is of little interest to anyone, or that it is not serving the purpose you intended. With solid statistics like these, you will be able to judge which sections can be removed or changed.

Download activity for the current period (assuming that you entered a previous period filename, otherwise it is for the entire life of the system) is shown next. This lets you see what files are hot and what files are not.

The user file statistics and miscellaneous message base statistics appear next. These are often useful in alerting you to unintentionally SYSOPed members or locked messages. The grand total "File Section Statistics" wrap things up, showing you how much of your disk space is being used for the file transfer section and how much of that is downloaded (total bytes).

While the information is mainly for your own use, many SYSOPs include this as a Menu option to allow their users to see how the board is being utilized. If you choose to permit your users to view it, include a line in a menu such as the following:

item = B, "B: BBS Statistics", 10, -, 108, STATS.TXT



## BBSFIX.COM

This program will repair almost any damage to your files that is non-physical. If you have set up your BBS to open all files when the program loads initially (see Setting Node Defaults), a power outage will cause the system to shut down with the files open. With other BBS programs, you would need to spend a considerable amount of time manually inspecting each file, and repairing it. Indexes and pointers need to be checked as well, and if a user were leaving a message at the time the power failed (worst case), numerous problems would result.

Running BBSFIX repairs all these damages automatically. To run the program, enter:

BBSFIX

You will see the following:

BBSFIX - BBS file rebuild/recovery - 4.0  
Copyright (c) 1985, Micro-Systems Software Inc.

General rebuild or Message links (G/M)?

If you select G, you will be asked:

Do you want to force rebuild (Y/N)?

A general rebuild examines the following files:

MSGHEAD.DAT  
MSGKEY.DAT  
MSGTEXT.DAT  
USERDESC.DAT  
USERKEY.DAT  
UDHEAD.DAT  
UDKEY1.DAT  
UDKEY2.DAT  
UDKEY3.DAT  
CALLER.DAT  
CALLKEY.DAT

If you do not "force" the rebuild, the program will repair only those files that are marked as changed or modified. (Simply because a file is not marked as changed or modified does not mean that it is not damaged, it just means that BBSFIX doesn't THINK that it is damaged. This is why you can force the system to rebuild all files.) If there are no such indications in a file, you will see the message:

No data file rebuild necessary.

If there is an error, you will see:

Rebuilding data file.

or

Rebuilding index file with batch size = 2999

or some other appropriate message. Other messages will appear, such as "d", "B1," "B2," etc. These should be of no concern to you; they are simply messages telling what the program is doing at the moment.

Forcing the rebuild with the Y option will examine every file, rebuild pointers, indexes and end of file markers as well as the open file flag on every one of the files whether it is damaged or not. As it examines each file, it will display a message telling how many entries are in the file. When it has examined and repaired each file, you will see the message:

End Of Rebuild.

The Message Link rebuild checks only the MSGHEAD.DAT, MSGKEY.DAT, and MSGTEXT.DAT files. This should be used in the event that the general file rebuild reports an error in one of the "MSGxxxx" files. The message link rebuild will restore the threads, relink, and renumber the messages.

BBSFIX has two parameters that may be entered:

-c:xxxx  
-m:xxxx

These allow you to set the number of callers and to renumber the message base. This is particularly helpful if you have updated from 3.04 or have converted an RBBS system. Either parameter may be entered separately, or both may be entered at the same time:

BBSFIX -c:12345 -m:9234

This will display the title and copyright notice, and then show:

Updating caller log file.  
Updating message file.  
End Of Rebuild.

(The structure of all files is discussed in Section VII.)

## BBSFILE.COM

This utility facilitates installing files in the download area. Unlike Function 18 (Upload a file from local, Section III), this allows you to install many files rather than single files. You can copy all the files on a disk to your download directory, then install them from DOS. Several parameters can be used:

-s:x      Section Number  
-d:x      Directory Number  
-t        Text Files  
-n        No Description  
d:\path\   Optional drive and path name  
wildmask   Consider only matching files

Entering BBSFILE without any parameters will assume binary files and enter them in Section 0, and Directory 0. It displays the title and copyright notice, then scans every file in the directory:

BBSFILE - BBS-PC! Local file utility - 4.0  
Copyright (c) 1985, Micro-Systems Software Inc.

[ \_\_\_\_\_ ]  
CENTURY.EXE  
MULTRAIN.COM  
SENATE.TXT  
FRKLBRY.COM  
TREKTRAK.EQE  
DIRUTIL.LQR  
JOYCE.DOC  
MURDRFIL.LBR  
LTNSLAGR.COM  
UNBOOBOO.EQE

[ \_\_\_\_\_ ]  
FINDDUPE.LQR

The files are displayed one at a time, and pause for you to enter the description. The dashed line is 40 spaces long, and is the maximum length allowable for the description; this appears before each ten files. As each filename is displayed, enter the description:

[ \_\_\_\_\_ ]  
UNBOOBOO.EQE

File recovery for platter crashes.

Because these files are assumed to be binary, when you come to a text file simply press ENTER and it will be skipped. You may also skip any files you want to install in a different directory, or a different section. After the last file has been displayed, a message will tell you how many files were installed.

If you have a number of text files to be installed in Section 5, Directory 11, for example, enter:

```
BBSFILE -s:5 -d:11 -t
```

Files may also be entered without a description. This does a mass upload of the files and lets the SYSOP add the descriptions with the Move option of the Browse command later.

```
BBSFILE -s:1 -d:2 -n
```

Wildmasks and paths may also be used. The pathnames would be used to tell BBSFILE where the files to be uploaded are located. Remember, you must run BBSFILE while you are logged into the main BBS directory. If the files were on DF1:, for example, the command might be:

```
BBSFILE df1: -s:1 -d:2 -n
```

The wildmasks are BBS-PC!'s own wildmask, not those of AmigaDOS. There are two wildmask characters:

- + Skip a single character
- \* Skip any number of characters

These are followed by a pattern. For example:

- \*.bas       Matches with ".bas" anywhere in the filename.
- ++p4       Matches with "p4" anywhere AFTER the first two characters.

You may also skip the wildmask characters altogether. In which case, the pattern must begin at the first character of the filename. Examples:

- prof       Matches with "profile" and "professor", etc.

These are, by the way, the same wildmask conventions used by the Catalog and Browse commands within the BBS.

## CHKFILE.COM

CHKFILE is designed to cross-reference files in the BBS upload/download database with files that are actually online. Two basic modes of operation are available depending on what parameters are specified. The first mode which is the default mode is designed to scan the database sequentially and test file online status. A display similar to the following is shown:

```
PAKMAN.EXE -> PAKMAN.EXE
123.EXE -> \BBS\UTIL\123.EXE
GAME/CMD -> \BBS\TRS80\FILE0001.U1
STARTREK.EXE -> STARTREK.EXE
STARTREK.DOC -> * Not Online *
ATARI BBS -> \ATARI\FILE0002.U1
```

Each entry in the database is shown along with the path the file is located in. If pseudo filenames are in effect for a particular section, the actual local filename shown. Files not found are indicated as "\* Not Online \*". There is switch that may be specified to delete any files that are not found and is specified with "-d" when CHKFILE is entered. If this option is used the message "\* Not Online \* - Killed" is indicated for files not online and the entry is deleted from the upload/download database.

The second mode of operation is engaged when a drive or wild mask is specified when CHKFILE is entered (see BBSFILE for a detailed description of these wildmasks). For example:

```
CHKFILE b:
CHKFILE *.BAS
```

This mode is inverse of the first, i.e. files in a specific directory are checked to see if they have an entry in the upload/download database. A path must also exist to any files specified or the message "% File not found in U/D path list %%" is outputted and the procedure is aborted. If no entry exists for a file it is displayed and CHKFILE moves on to the next. The "-d" switch is also valid for this mode but has a slightly different meaning. If used this option will cause CHKFILE to pause on files not found in the database and prompt for a deletion like so:

```
PAKMAN.EXE Delete (Y/N)?
STARTREK.EXE Delete (Y/N)?
```

If 'Y' is entered at this point the file will be deleted. A 'N' response will cause no action to be taken.

One other point that should be noted is pseudo filenames created by the system such as "FILE0001.U1" will not be seen as entries in the database since they exist as an alias. Care should be used before deleting these files.

## Part V

### Principles of Operating a BBS

The world of bulletin board systems is a fascinating one. Others will be able to communicate with you and with each other. Lasting friendships have developed between people who have never met in person. Marriages have resulted from these friendships, and so have divorces! Being a SYSOP is a unique experience. As a SYSOP you will earn the friendship (or enmity) of your users depending on how you operate the board, as well as on your own personality. As you contend with "crashers" (people who delight in disrupting BBS operation to aggravate you and your users), you will find satisfaction in being able to defend your system.

On the negative side, there is the expense of the equipment and supplies, as well as the daily effort of maintaining the board. While BBS-PC! is a maintenance-free as any BBS can be, you will find yourself spending some time each day in routine maintenance, adding new files to your download section, changing menus, and so on. As one experienced SYSOP said, "We must be a bunch of masochists. We've invested several thousand dollars in equipment, dedicated phone lines, give up the pleasure of being with our families or taking in a new movie to spend the time maintaining and improving our systems just to allow total strangers to come into our homes and play with all of it, then do his best to destroy it and laugh at us. Sometimes I think it's safer to invite Attila the Hun to use my home for field maneuvers."

Despite this, new BBSes spring up every day. The negative aspects of running a BBS also result in the rapid disappearance of many of the newcomers. The discovery of public bulletin boards by users who acquire modems for the first time brings a new outlook on life, new horizons, new ways to spend leisure time. Having used several boards, an inexperienced BBSer is lulled into thinking, "This look like it would be fun," and he decides to set up a board. He gives little or no thought to why anyone would want to call his board, and attempts to offer the same features as the ones he has found in his limited journey on the telecommunications circuit. Some of the more creative ones write their own programs, and very decent ones. Others spend weeks or months at their programming, abandon the project and search for an existing program, spend additional time changing it, and then go on-line. After several days or weeks of wondering why callers are few and infrequent, they discard the project. One set up a board which was quite popular with the local teenagers, and after six months had one of the most popular boards in his area. He came home from work one evening and found his entire message base wiped out, without a clue how it had been done. He learned that there was no protection from anyone who used the "Kill" command to kill any message in the system, he was ready to throw in the towel. Fortunately for the dozens of young people who had become loyal users, he was able to correct the problem, although the messages were gone.

The most important thing for any public BBS is that it have a purpose for existing. Simply "setting up a BBS" is not enough to guarantee its success. Unless you can fill a real need, or offer your users something they cannot find anywhere else, they will stop calling almost as soon as they find you. Some very successful systems are operated by User Groups, and serve as a place to exchange ideas, hardware information, programming tips, and software. Such boards are fairly common, but as long as they serve a real purpose for their users they will enjoy continued success. Some boards serve no other purpose than public message bases. The oldest BBS in the West Palm Beach, Florida, area, has been in continuous operation for more than six years. It offers no downloading or uploading, only one game to play (if you can solve the riddle, you get a free box of disks), and no private messages. It operates today, as it did six years ago, at 300 baud, but still has loyal users.

When the author of this manual had the idea of establishing a BBS for other writers, he was told that writers across the country would never call a BBS in Florida for news of markets in New York or California. He carries no games or programs in the download, only text files of market information of interest to writers, editors, and others in the publishing industry. The board is also operated on a one-drive, 256K PCjr. Despite the seemingly overwhelming odds against success for such a BBS, it has been in operation for almost three years, and includes top-notch journalists and award-winning fiction writers among its members, who call from 36 states. That board, like other successful ones, is filling a need for its users. The proof that even with such a restricted purpose a BBS can fill a need is that two of the users have reported finding new markets for their articles they earned more than \$3,000 within the first year they began calling. One has not only increased her sales, she has been named contributing editor to a monthly magazine that is known to most computerists as a result of finding a request for new writers.

Find the need and fill it, and you'll have a successful BBS.

BBS-PC! removes nearly all the work of maintaining a BBS. "Nearly all," because every system needs regular attention. You are free to use it for almost any type of communications center you wish. You can configure it to be a totally restricted, in-house message center, for example, and have it verify the caller by disconnecting after he has entered his name and password, then calling back at the number listed in his records. All users must be given even minimum access by the SYSOP, and are strictly limited in what they may do while on the system. Throughout this manual we have made reference to the "Widget Company, and show how BBS-PC! can function as a powerful message center for a sales organization. Naturally, a board of this type will require a fair amount of time to maintain.

At the other extreme, BBS-PC! can be run as a totally open system, allowing members to add themselves, post and retrieve mail, upload and download files at will. Once a maximum number of messages has been entered into the system, there is no danger of a "crash" because another message is entered; in this case, the lowest number is simply scrolled off, releasing the space for additional messages. Because of the ingenious programming, there is never a need to pack the message base; when a message is deleted,

the space is immediately flagged as being available for use again. It is conceivable that such a system could be operated with only a once or twice weekly maintenance session by the SYSOP.

Most systems will fall somewhere in between the two extremes. Let's take a look at two totally different ways to operate a BBS. We'll call these the "Open" system and the "Restricted" system. A third type is the "Closed" system, which is actually a private message center, and usually found within the corporate environment.

### OPEN Systems vs. RESTRICTED Systems

Although there are many possible combinations of access levels, it essentially means you must choose one of two ways to run your board. Either it will be an "open" system, where any caller can become a member just by using a command to add himself, or it will be a "restricted" system, where only the SYSOP may add users.

### Defining Your Sections

An extremely important aspect of getting your bulletin board on line is designing your section structure. This involves telling BBS-PC! what sections Guests may access and what sections Members may access. In addition, you may define how they can access all of the various sections. This will do much to determine the atmosphere of the board. To set the board's section structure, use the "S" command at the SYSOP maintenance menu.

Each message in the message base or file in the file transfer area carries a section number with it. There are 16 sections, numbered 0-F. Each user may have access to as many or as few of these as the SYSOP desires. If a user does NOT have access to a section, he will never see messages or files in that section. This allows you to restrict users to certain areas of the system.

As an example, you may decide that section 6 should become the IBM PC area. You would only enable for access to section 6 such users as you want (obviously, IBM PC users); only users with section 6 access would see messages or files stored in this area.

By carefully allocating sections and not giving everyone access to everything, you can have the appearance of several boards in one. Even though all of the messages and files exist side by side in the system, users with access to different sections will think they are on totally different boards.

In addition to being able to configure each user's section access, you may control what KIND of access the user has to that section. Each user may be configured separately for read, write, upload, and download access to each section. There are many applications of this. One is that the SYSOP may make special "write only" sections into which callers send messages and files. The SYSOP can then transfer the data to sections where it can be accessed. This



effectively protects SYSOPs from illegal information being sent into the system without their knowledge, and provides protection for his Members in the event an obnoxious caller decides to leave obscene graffiti.

In another application, the SYSOP may allow users to read and write messages in a particular section, but not allow them access to files in that section of the U/D menu. This permits the SYSOP to define several sections for the user mail, but limit file transfer.

The user is able to see what sections he may access, and what kind of access it is, by using the "N" command from the main command prompt.

There are two kinds of callers, Guests and Members; each of these can have separate "default section access levels." With two different classes of callers, each having up to four different kinds of access into 16 different sections, you can see how this provides for many possible combinations.

The program will display, insofar as possible, explicit messages when a user is denied access to a section. For example, if a user attempts to reply to a message that is in a section to which he does NOT have write access, he will see "Write access denied" when he attempts to store the message. Unless he is unusually dense, he will understand that he can READ from this section, but not WRITE to it.

Also, if a caller's access is not sufficient to operate a specific command, that command will not even be shown on the menu. Any attempt to use a command without sufficient access causes the system to respond exactly as if that command did not exist.

For the most part, BBS-PC! attempts to display very specific error messages. An error message surrounded in percent signs is a system error message, placed there for diagnostic purposes by the author. Usually, these messages are the result of moving system files without changing path names with BBSINIT. Missing text files are another common mistake. Make careful note of the message, and refer to the section of the manual which contains a list of all error messages and their descriptions.

As mentioned earlier, BBS-PC! allows you to define default section access levels for both Guests and Members. If you wish, you may set both of these to identical levels. But most SYSOPs prefer to have Guests with access to only a few sections, and not permit them to leave public mail at all. This can be used to require them to take the time to become a member prior to using the system.

Let's take two examples. One is an open system, with little restriction on callers, and the other a restricted system that controls Guests tightly. The first system allows any caller to send a message or file into a section where the general calling public can see it. The second system permits Guests to leave no public mail, and allows only protected uploads. (Protected uploads are those which may not be seen or downloaded by anyone until the SYSOP has approved it.)

## The OPEN System

In our first example, we will allow Guests to have read, write, upload, and download access to sections 0-3. Members have read, write, upload, and download access to sections 0-7. Sections 8-F may then be used for various special purposes, and users are enabled for each of these as needed.

A New User calling this system might see this menu:

```
+-----MAIN MENU-----+
| B: Bulletins and information |
| C: Chat with the SYSOP |
| E: End this call (goodbye) |
| F: File transfer menu |
| L: Leave a message |
| M: Mask out unwanted sections |
| O: View/edit your user options |
| P: Private EMAIL menu |
| R: Read messages |
| S: Scan message headers |
| T: Time used during this call |
| U: Display recent callers |
| X: Toggle expert menus on/off |
| Y: Show your caller statistics |
+-----+
A: Add yourself to user files
```

Main Func (? for menu):

When a caller logs in to the system for the first time, he would be able to read and leave messages in sections 0-3. Also, he would be able to download files from those sections. All of the Members of the system would have access to sections 0-7. More than likely, you would want to keep most of the material in section 0 general in nature. Any information you want limited to Members only gets stored in any section above 3.

All a Guest needs to do to become a Member in this system is use the "A" command at the main menu. Your non-member bulletin should instruct him to do this, or at least direct him to the membership information file that is on the bulletin sub-menu. As soon as Guests use the "A" command, their status is updated and they have the additional access to sections 4-7.

To set up your sections, select the "Maintenance module" option at the Wait for Caller menu and press RETURN. When greeted by the SYSOP command prompt, type: S <RETURN>. The current section defaults will be displayed:

## Guests:

A: Time limit: 60  
B: Privilege : 1  
C: RD access : 0123  
D: WR access : 0123  
E: UP access : 0123  
F: DN access : 0123  
G: Default : 0

## Members:

H: Time limit: 120  
I: Privilege : 2  
J: RD access : 01234567  
K: WR access : 01234567  
L: UP access : 01234567  
M: DN access : 01234567  
N: Default : 1

The sections in this open system could be:

0: [DFTP—R] General Mail  
1: [D—\*—] Tech Tips  
2: [DFT—] Trivia!  
3: [DFTP—R] Personal Mail  
4: [DFTPB\*R] Members Only  
5: [DFTP—] The BAR Room  
6: [DFTP—] Hidden Valley Club  
7: [D—] Guru's Movie Reviews  
8: [DFTPB—] V.I.P. Lounge  
9: [—]   
A: [—]   
B: [—]   
C: [—]   
D: [—]   
E: [—]   
F: [—]

In the example above, Guests may leave private mail in section 3, and read anything or write in sections 0-3. Section 8 is available for the SYSOP to grant to users on a selective basis. Sections 0, 4 and 8 have bulletins to be displayed. Members who have been approved by the SYSOP have all of sections 0-7.

In the open system above, joining the system gives a Guest access to the additional sections 4-7, and that is the only advantage (which might be great, depending on what is contained in these sections). Guests are allowed to read mail, leave mail, upload files, and download files. They are just restricted to fewer sections.

```
0: [DFT-B-R] General Mail
1: [D-----] Tech Tips
2: [DFTP----] Trivial
3: [DFTP--R] Personal Mail
4: [DFTP*B-R] Members Only
5: [DFTP----] The BAR Room
6: [-----]
7: [-----]
8: [DFTPB--] V.I.P. Lounge
9: [-----]
A: [-----]
B: [-----]
C: [-----]
D: [-----]
E: [-----]
F: [-----]
```

In the example above, notice that section 0 does not allow private mail. Section 1 not only prohibits private mail, but also removes the addressee and author's name during a scan of the message headers. (Who needs to see anything other than a subject when scanning technical suggestions?) Section 8 is available for the SYSOP to grant to users on a selective basis. Sections 0, 4 and 8 have bulletins to be displayed. (NOTE: If you want bulletins for these sections, you must have not only BULL-0.TXT in the directory, you also need BULL-4.TXT and BULL-8.TXT.)

In the open system above, joining the system gives a Guest access to the additional sections 4-7, and that is the only advantage (which might be great, depending on what is contained in these sections). Guests are allowed to read mail, leave mail, upload files, and download files. They are just restricted to fewer sections.

In this open system, you will probably want to adjust the user log access range so that both Guests and Members are included. (If you remember, this was an option of the SYSOP command menu's "T" command.)

Also, note that both Guests and Members have a "default section" which you determine from the SYSOP menu, too. This is a convenience feature for the SYSOP. When a message is stored without a specific section number, it will be stored in the default section.

### The RESTRICTED System

In our second example, a restricted system, Guests will not be allowed to leave public mail, nor will they be allowed to upload a file to a section where any Members (other than the SYSOP, of course) can see it or download it.

To do this, Guests will be allowed write access to section 0, but only the SYSOP will have read access to that section. In the same manner, Guests will be allowed to upload to section 0, but only the SYSOP can see files that are in that section. In effect, section 0 has become "write only".

You should give Guests read access to at least ONE section. In our example system, Guests will be allowed read access to section 1. This means that Guests may read messages from either the SYSOP or other users, as long as these messages are stored in section 1. Guests will not be allowed access of any kind to any other section in the system. (NOTE: This means that they cannot download any files.) When a first time caller logs in, he might be greeted by a NEWUSER.TXT file something like this:

---

Welcome to the Hidden Valley Computer Club BBS!

It is a pleasure to have you visit HVCC BBS. This system is sponsored by the IBM PC user's group of Hidden Valley, VT, and is not a system open to the general public. As a Guest of this system, you will be allowed to read messages in the General Mail section only. Any messages you leave will be private to the SYSOPs. If you have a message for a member of the Hidden Valley Computer Club, you may leave it addressed to the member, and a SYSOP will forward it for you.

You may also upload files, if you wish, but these files will be accessible by the SYSOP only, until approved and posted for the membership. As a Guest, you will not be allowed to download any files.

For information on becoming a member of the Hidden Valley Computer Club, select option B at the main menu. From the Bulletin Menu, select option 4. This bulletin explains who we are, what we stand for, what we offer our members, and what we expect from our members. Also, you may leave a comment to SYSOP on the way out asking for information.

If you have any questions, the same bulletin sub-menu holds a help file that explains most of the commands. Thanks again for calling. To view this bulletin again, use option "1" from the Bulletin Menu.

### The SYSOPs

---

This non-member bulletin explains to callers clearly what they can and cannot do. Although it does not state that there are other sections they cannot see, it does give the impression there is more for members.

The Guests can read mail from section 1 left by you and other Members, but when they attempt to leave a message, it can only be stored in section 0. As the SYSOPs read each message, if it is of general interest or for a member of the system, they can forward this message into another section (changing the subject and to whom it is addressed, if necessary).

On the other hand, if the message is one that is filled with obscene or abusive language, or contains information of a nature the SYSOP does not wish to allow on the system, it is a simple matter to delete the message and no one except the SYSOP will ever read it.

Using a "write only" section permits the SYSOP to filter all messages before they are placed into the open board. This prevents an anonymous caller from placing any potentially improper information in an area where it can be seen. Another consideration is that when would-be system crashers run into BBS-PC!, they usually get frustrated because they find they are unable to annoy you or your users. Most callers of this type have a tendency to harass the users with silly or obscene messages. With a write only section for Guests, you can console yourself with the thought that you can delete messages MUCH faster than they can leave them.

User log access would also be set such that Guests do not appear. This prevents more inventive would-be crashers from abusing you simply by logging in with foul language for names. All callers are logged, but you set the privilege levels of the names that may be seen by users. (Set the low privilege, which is the first number, to a value higher than Guest privilege.)

Members in this system would not have read access to section 0. This prevents any Guests from leaving rude messages and disturbing Members. As you can see, the impetus to join this closed system is different than with the open system. In the closed system, you are severely limited in your actions until you become a Member.

Both Guests and Members in this system will be assigned upload access to section 0. Members will be allowed to download from sections 1-4. This has the effect of making a write only upload area, as well. Every time the SYSOP logs in, if new files have been uploaded, a message is printed. It is the SYSOP's job to review the files, and if they are acceptable, forward them to a section where they can be accessed (changing the description, if need be). Again, this offers the SYSOP a high level of protection from callers leaving unauthorized or illegal (copyrighted) materials in the download area.

For the restricted system we have been outlining, the system defaults would appear like this:

**Guests:**

A: Time limit: 60  
B: Privilege : 1  
C: RD access : 1  
D: WR access : 0  
E: UP access : 0  
F: DN access :  
G: Default : 0

**Members:**

H: Time limit: 120  
I: Privilege : 2  
J: RD access : 1234  
K: WR access : 1234  
L: UP access : 0  
M: DN access : 1234  
N: Default : 1

**Password Protecting Your SYSOP Account**

When you ran the BBSINIT program, it installed "SYSOP" as a user. Whether or not you choose to use this name in the system or not is up to you, but this account had to be used for the initial system setup. The SYSOP user account carries FULL access (i.e. privilege of 255 and access to all sections).

BBS-PC! has a unique protection feature in that a Section SYSOP (a regular member, but one who has SYSOP privileges for maintaining one or more sections) may not edit the record of another user if the second user's record has a higher privilege than the first user's. Similarly, a user cannot assign a section to, or remove a section from, another user if he does not have that section present in his own mask. In installations with multiple SYSOPs, this becomes very important. At least ONE account must carry the highest possible access in all areas, or there would not be any method of assigning these access levels to others. We referred to this as the Chief SYSOP or a Wizard SYSOP.

**Keep Your Users In Mind**

No bulletin board will be successful if it treats its callers callously. Any businessman knows that without the customer who walks through the door he is out of business. As a SYSOP you'll be out of business if you don't consider your callers just as important as the customer walking through the door. If you antagonize them, or make it difficult for them to use your system, they will quietly (sometimes not so quietly) move on to more pleasant boards.

Give them adequate help in using the board correctly. We could have included a section on "How to Leave Messages" and another on "How to Read Messages." BBS-PC! is so versatile, and SYSOPS so imaginative, that such sections would become meaningless. On one system you "Leave" a "Message," on another you "Enter" a "Bulletin," and on yet another you "Post" some "Mail." Private messages don't exist on some, and on others the user has no choice of which section in which to leave a message (or post his mail, or enter a bulletin). Because of this, you will have to write your own help files (using those supplied as a very skimpy suggestion). You may want to have several help files for leaving messages, reading messages, using the threads, entering a SIG, etc. Simply make all the helps available at the menu where they are most needed, and again at a "Help Menu," if you think this would be advantageous to your users. The number of menus and text files you make available to your users is limited only by your disk space and your imagination.

By all means, pay attention to requests you get from your users. While you do not have to act positively on all of them, you never know when one of those suggestions might be just the thing that will set your board apart from all the others. Many of the features of BBS-PC! that make it the ultimate in bulletin boards have come from users. While BBS-PC! was undergoing nearly a year of development and beta testing, the users told us what they wanted. Even if we didn't think it was a "hot" suggestion, if we saw it often enough, it found its way into the program you now have in your hands.

Your users are your only reason for having a BBS. Listen to them; if you don't, you'll find yourself with a phone that never rings.

Karl H. Meyer  
SYSOP, The Notebook



THIS PAGE INTENTIONALLY LEFT BLANK

## Appendix A

## Technical Details and Error Codes

## File Structures

This section is intended to be a brief technical description of the file structures for BBS-PC version 4. This information is required for writing support programs that interface with the BBS software. Knowledge of the 'C' programming language will be helpful.

BBS-PC consists of the following data files:

| Name         | Rec Length | Description                     |
|--------------|------------|---------------------------------|
| BBS.P        | --         | General purpose parameter file  |
| CFGINFO.DAT  | --         | Node independent configurations |
| NODExx.DAT   | --         | Node dependent configurations   |
| MSGKEY.DAT   | --         | Keys for messages by number     |
| MSGHEAD.DAT  | 114        | Message headers                 |
| MSGTEXT.DAT  | 256        | Message texts                   |
| USERKEY.DAT  | --         | Keys for user records by name   |
| USERDESC.DAT | 120        | User records                    |
| UDKEY1.DAT   | --         | Keys for files by name          |
| UDKEY2.DAT   | --         | Keys for files by directory     |
| UDKEY3.DAT   | --         | Keys for files by age           |
| UDHEAD.DAT   | 108        | File headers                    |
| CALLKEY.DAT  | --         | Keys for caller by number       |
| CALLER.DAT   | 72         | Call records                    |

Due to the complexity and proprietary nature of the keyed access routines, No attempt will be made to explain their contents except to say a B-TREE index structure is used. If your application demands further explanation please contact Micro-Systems Software directly.

BBS.P is a general usage parameter file which contains all the indexed files, record sizes and path requirements for opening and access. This is a standard ASCII file and can be edited/changed as necessary although not recommended.

All data files except BBS.P, CFGINFO.DAT and NODExx.DAT consist of a 128 byte header record followed by records of length which is determined by contents. The first data record is positioned on a multiple of its length and not necessarily at position 128 (just past the header). When a record is deleted it is simply marked as such and added to the chain of deleted records. New records are taken from the chain of deleted records until exhausted and more space will be allocated to a file as necessary. Infinite growth is controlled by variables maintained in CFGINFO.DAT. Once the "maximum" size is reached two paths are taken by the software based on the type of file. For the message and caller files the oldest record is internally deleted and the new record replaces it. For all remaining files an error condition is issued and external steps must be taken to correct the situation.

The following 'C' structures contain references to the following data types and should be interpreted as follows:

|        |                          |
|--------|--------------------------|
| char   | signed 8 bit interger    |
| byte   | unsigned 8 bit interger  |
| short  | signed 16 bit interger   |
| bits   | unsigned 16 bit interger |
| ushort | unsigned 16 bit interger |
| long   | signed 32 bit interger   |

Integers are stored in standard 8088 format (low to high) and data alignment is explicitly shown. Bit fields are assigned from 0-15 but have been padded to force alignment to bit 15.

#### Miscellaneous constant definitions

```
#define NUM_TERM 10 /* computer types */
#define NUM_LINES 2 /* number of phone lines */
#define NUM_DIR 15 /* number of phone directories */
#define NUM_SECT 16 /* number of section names */

#define FPATH_LEN 30 /* file name path lengths */
#define FNAME_LEN 12 /* file name lengths */
#define NAME_LEN 24 /* user name length */
#define LOC_LEN 24 /* user location length */
#define PHONE_LEN 12 /* user phone length */
#define PASS_LEN 10 /* user password length */
#define CAT_LEN 15 /* file catalog names length */
#define DESC_LEN 40 /* file catalog desc length */
#define FROM_LEN 24 /* message FROM field length */
#define TO_LEN 24 /* message TO fields length */
#define SUBJ_LEN 24 /* message subject length */
#define MSG_LEN 2000 /* message length */
#define TERM_LEN 15 /* computer name lengths */
#define SECT_LEN 20 /* section name lengths */
#define CATG_LEN 20 /* directory names lengths */
#define LINE_LEN 15 /* phone line names length */
#define PNAME_LEN 15 /* phone directory name lengths */
#define PNUMB_LEN 25 /* phone directory number lengths */
#define MODEM_LEN 30 /* modem ipl string len */
#define DIAL_LEN 20 /* modem dial prefix/suffix len */
```

```
/* COMM parameters
*/
typedef struct {
 ushort : 10; /* reserved */
 ushort parity: 1; /* comm - parity 1 = on */
 ushort even : 1; /* comm - parity 1 = even */
 ushort stops : 1; /* comm - stops 0-1 */
 ushort word : 3; /* comm - word size 0-3 */
} PARAMS;

/* terminal parameters
*/
typedef struct {
 ushort : 15; /* reserved */
 ushort linefeed: 1; /* linefeeds */
 byte nuls; /* nuls */
 byte protocol; /* protocol */
 byte page[2]; /* page size */
 byte bs[3]; /* BS codes */
 byte cls[3]; /* CLS codes */
 char name[TERM_LEN+1]; /* terminal name */
} TRMNL;
```

```
/* CFGINFO.DAT record structure
```

```
*/
```

```
typedef struct {
```

```
 ushort : 14; /* reserved */
 ushort dir0_ok: 1; /* Directory 0 downloads */
 ushort by_call: 1; /* time limit per call */

 short max_msg; /* maximum messages in system */
 short max_user; /* maximum users in system */
 short max_log; /* maximum call log */
 short max_ud; /* maximum U/D files */
 short reward; /* upload reward */
 short t1; /* reserved */
 short t2; /* reserved */
 short t3; /* reserved */
 long sleeptime; /* sleep timeout (adjusted) */
 byte log_pl; /* log privilege low */
 byte log_p2; /* log privilege high */
 byte t4; /* reserved */
 byte hi_men; /* highest menu set */
 short log_type; /* login method */
 short limit[2]; /* guest/member time limits */
 short priv[2]; /* guest/member privileges */
 bits rd_acc[2]; /* guest/member read access */
 bits wr_acc[2]; /* guest/member write access */
 bits up_acc[2]; /* guest/member upload access */
 bits dn_acc[2]; /* guest/member download access */
 byte sav_sec[2]; /* guest/member save section */
 byte sec_flg[NUM_SECT]; /* section flags */
 char sec_name[NUM_SECT] [SECT_LEN+1]; /* section names */
 char ud_alt[NUM_SECT] [FPATH_LEN+1]; /* alternate UD paths */
 char syspass[PASS_LEN+1]; /* SYSOP password */
 byte menu[2]; /* guest/member menu sets */
 byte align;
```

```
 TRMNL trmnl[NUM_TERM]; /* terminal parameters */
```

```
) CFGINFO;
```

```
/* NODExx.DAT record structure
```

```
*/
```

```
typedef struct {
 ushort : 12; /* reserved */
 ushort frame : 1; /* frame on/off */
 ushort bell : 1; /* bell on/off */
 ushort wakeup : 1; /* wakeup on call */
 ushort opens : 1; /* file opens type */

 bits line_mask; /* rs-232 lines mask */
 bits poll_mask; /* poll mask */
 ushort serial; /* upload serial number */

 short mbaud[NUM_LINES]; /* modem default bauds */
 byte mtype[NUM_LINES]; /* modem result types */
 char modem[NUM_LINES] [MODEM_LEN+1]; /* modem ipl codes #1 & #2 */
 char dialp[NUM_LINES] [DIAL_LEN+1]; /* modem dial prefix */
 char dials[NUM_LINES] [DIAL_LEN+1]; /* modem dial suffix */
 char result[NUM_LINES] [3] [MODEM_LEN+1]; /* 300/1200/2400 strings */

 byte dir_lines[NUM_DIR]; /* phone directory lines */
 short baud[NUM_DIR]; /* phone directory bauds */
 byte align;

 PARAMS com[NUM_DIR]; /* phone directory params */
 char dir_names[NUM_DIR] [PNAME_LEN+1]; /* phone directory names */
 char dir_numbers[NUM_DIR] [PNUMB_LEN+1]; /* phone directory nums */

 ushort t1; /* reserved */
 ushort t2;
 ushort t3;
 ushort t4;
 ushort t5;
} NODE;
```

```
/* MSGHEAD.DAT record structure
```

```
*/
```

```
typedef struct {
 byte type; /* record type */
 ushort : 11; /* reserved */
 ushort forward : 1; /* message forwarded */
 ushort readbyto : 1; /* message read by to person */
 ushort personal : 1; /* personal message flag */
 ushort unformat : 1; /* unformatted message */
 ushort locked : 1; /* message locked */

 long number; /* message number (key) */
 byte section; /* section number */
 char from[FROM_LEN+1]; /* from field */
 char to[TO_LEN+1]; /* to field */
 char subj[SUBJ_LEN+1]; /* subject field */
 ushort recno; /* data record number */
 ushort date; /* date (packed) */
 ushort time; /* time (packed) */
 long origin; /* message origin (if a reply) */
 long replys[5]; /* message replys */
 byte align;
} MSGHEAD;
```

```
/* MSGTEXT.DAT record structure
```

```
*/
```

```
typedef struct {
 byte size; /* text size (255 invalid) */
 ushort recno; /* linked data record */
 char text[253];
} MSGTEXT;
```

```
/* USERDESC.DAT record structure
```

```
*/
```

```
typedef struct {
 byte type; /* record type */
 ushort : 12; /* reserved */
 ushort callback: 1; /* requires call-back */
 ushort guest : 1; /* guest access */
 ushort expert : 1; /* expert menus */
 ushort linefeed: 1; /* linefeed flag */

 char name[NAME_LEN]; /* user name (key) */
 char loc[LOC_LEN+1]; /* location */
 char phone[PHONE_LEN+1]; /* phone number */
 char pass[PASS_LEN+1]; /* user password */
 short limit; /* limit time (mins) */
 short time_rem; /* time remaining for day (mins) */
 short upl_total; /* total uploads by user */
 short dnl_total; /* total downloads by user */
 short msg_total; /* total messages left by user */
 short calls_total; /* total calls made by user */
 long high_msg; /* high message number */
 ushort date; /* date last called (packed) */
 bits bul_flg; /* bulletin flags */
 bits sys_acc; /* sysop access */
 bits rd_acc; /* non-member sec read access */
 bits wr_acc; /* non-member sec write access */
 bits up_acc; /* non-member upload access */
 bits dn_acc; /* nonmember download access */
 byte term; /* terminal type */
 byte menu; /* menu set */
 byte priv; /* privilege number */
 byte sav_sec; /* save section */
 byte protocol; /* up/down protocol */
 byte width; /* display width */
 byte length; /* page length */
 byte nuls; /* required nuls */
 byte cls[3]; /* clear screen codes */
 byte bs[3]; /* backspace codes */
} USRDESC;
```



```
/* UDHEAD.DAT record structure
```

```
*/
```

```
typedef struct {
```

```
 byte type; /* record type */
```

```
 ushort : 14; /* reserved */
```

```
 ushort local: 1; /* local file */
```

```
 ushort bin : 1; /* binary file */
```

```
 char cat_name[CAT_LEN]; /* catalog filename (key1) */
```

```
 ushort date; /* upload date serial */
```

```
 byte dir; /* directory number (key2 - seg1) */
```

```
 byte section; /* section number (key2 - seg2) */
```

```
 ushort accesses; /* number of accesses */
```

```
 long length; /* file length */
```

```
 char disk_name[FNAME_LEN+1]; /* disk filename */
```

```
 char owner[NAME_LEN+1]; /* owner's name */
```

```
 char desc[DESC_LEN+1]; /* description text */
```

```
 byte align;
```

```
} UDHEAD;
```

```
/* CALLER.DAT record structure
```

```
*/
```

```
typedef struct {
```

```
 byte type; /* record type */
```

```
 long number; /* caller number (key) */
```

```
 byte node; /* node number */
```

```
 short baud; /* baud rate */
```

```
 PARAMS com; /* COM parameters */
```

```
 char name[NAME_LEN+1]; /* caller name */
```

```
 char loc[LOC_LEN+1]; /* caller location */
```

```
 byte priv; /* callers privilege */
```

```
 byte disconnect; /* disconnect type */
```

```
 ushort in_date; /* log-in date (packed) */
```

```
 ushort in_time; /* log-in time (packed) */
```

```
 ushort out_time; /* log-out time (packed) */
```

```
 byte dnls; /* downloads */
```

```
 byte upls; /* uploads */
```

```
 byte msgs; /* messages */
```

```
 byte align;
```

```
} USRLOG;
```

## Error Codes

The following is a list of the most common error codes you might encounter. This list is not intended to be complete. If you should encounter an error not listed here please contact Micro-Systems Technical Support.

| Code | Description                                    |
|------|------------------------------------------------|
| 10   | Not enough memory for initialization           |
| 12   | Could not open index file                      |
| 13   | Could not open data file                       |
| 14   | Index file appears corrupt at open             |
| 15   | Data file appears corrupt at open              |
| 16   | Could not create index file                    |
| 17   | Could not create data file                     |
| 18   | Tried to create existing index file            |
| 19   | Tried to create existing data file             |
| 24   | Could not close index file                     |
| 25   | Could not close data file                      |
| 30   | End-Of-File encountered                        |
| 31   | Delete chain not pointing to deleted record    |
| 35   | Seek error, disk possibly full                 |
| 36   | Read failure on record                         |
| 37   | Write error, disk possibly full                |
| 42   | Duplicate record lock                          |
| 102  | Missing ISAM parameter file (BBS.P)            |
| 103  | Could not read parameter file initializers     |
| 106  | Could not read parameter file data descriptor  |
| 111  | Could not read parameter file index descriptor |
| 112  | Pending locks found in two-phase protocol      |
| 113  | Too many locks held by system                  |
| 114  | Active data record found deleted               |
| 500  | Could not open raw file                        |
| 501  | Could not close raw file                       |
| 502  | Could not read from raw file                   |
| 503  | Could not write to raw file                    |

THIS PAGE INTENTIONALLY LEFT BLANK

## Appendix B

### Special Modem Section

#### Introduction

This is the BBS-PC! modem section. It contains detailed setup instructions for users of the Hayes Smartmodem, and it contains concise summaries for some other modem types. If, after reading through this section, you still cannot make your modem work, contact Tech Support.

To those of you not using Hayes modems, you should be relieved to know that BBS-PC! does everything in its power to support command driven modems. With 16 different modem types and programmable setup strings, this system is extremely flexible.

Periodically, we will acquire new modems and add to our experience. If you have a modem that is not in this section, contact Tech Support. (Please remember that if we have not seen the modem, we may not be able to help you.)

BBS-PC! will work with any modem that supports the following:

- #1. It must report carrier ONLY when a caller is present.
- #2. It must hang up the telephone line either when the DTR signal is turned off, or when "true break" is transmitted.
- #3. If you want the terminal module to auto-dial, the modem should support dialing with ASCII command strings.
- #4. It must answer the telephone and establish carrier automatically (i.e. a true "auto-answer" modem).
- #5. For automatic detection of incoming baud rates, the modem must report the speed of the incoming call to the computer in some manner. (For some 300/1200 baud modems, this can be a signal on pin 6 - DSR of the computer. With 300/1200/2400 baud modems, you can program result codes into the modem defaults and BBS-PC! will scan for these.)

Although we discuss the details of setting the Hayes Smartmodems up for BBS-PC! here, additional modems will be discussed later in this addendum. If your modem is "Hayes compatible", then some of the information here should apply to you as well.

The Hayes Smartmodem 300 and Smartmodem 1200 have a series of DIP switches under the front panel of the modem. You should set the in the following manner:

| Switch | Position |
|--------|----------|
| 1      | UP       |
| 2      | UP       |
| 3      | UP       |
| 4      | DOWN     |
| 5      | UP       |
| 6      | UP       |
| 7      | UP       |
| 8      | DOWN     |

To set up the Hayes Smartmodem 2400 to operate with BBS-PC!, you must have a terminal program of some sort (such as our own MTERM or PC-TALK). You will have to enter a series of AT commands to the modem. The procedure is as follows: boot up the terminal program and set the transmission speed to 2400 baud. It does not matter whether you are at 7 bit word, even parity or 8 bit word, no parity. But the baud rate must be 2400.

Type the following sequence of commands to the modem:

| Command          | Meaning                 |
|------------------|-------------------------|
| ATBO <ENTER>     | CCITT mode              |
| AT&C2 <ENTER>    | True carrier detect     |
| AT&D1 <ENTER>    | Hang up on DTR          |
| ATEOQ1V1 <ENTER> | No echo, quiet, verbose |
| ATSO=1 <ENTER>   | Answer phone on 1 ring  |
| AT&W <ENTER>     | Write to permanent RAM  |

Now you have your hardware set up correctly. BBS-PC! can configure a modem setup string that will be sent to each modem whenever the program loads. With the Hayes Smartmodem and other compatible units, this allows you to use the AT command set to configure certain items that cannot be set permanently with the DIP switches, through software, each time you load the program. The default baud rate and modem type should also be set.

Under modem defaults, the first item listed is the default baud. This specifies the baud rate at which the setup string will be sent to the modem. (In the case of type "1" and type "3" modems, it also specifies the baud rate at which BBS-PC! will expect the modem to return result codes.) Obviously, if your modem is 300 baud only, this should be set for "300".

For the Hayes Smartmodem 300, set this to "300". For the Smartmodem 1200, set this to "1200". For the Hayes Smartmodem 2400, set it to "2400".

The next item is the modem type. This breaks down according to the following table:

## Type Function

- 0 Turns off the DTR signal to hang up the telephone, and assumes that result codes will be returned at the same baud rate as the last connection.
- 1 Turns off the DTR signal to hang up the telephone, and assumes that result codes will be returned at whatever the default baud rate is set to.
- 2 Transmits "true break" to hang up the telephone, and assumes that result codes will be returned at the same baud rate as the last connection.
- 3 Transmits "true break" to hang up the telephone, and assumes that result codes will be returned at the default baud rate.
- 4 Turns off the DTR signal to hang up the telephone, and assumes that result codes will be returned at the same baud rate as the last connection. Each time DTR is turned off, BBS-PC! sends the modem the setup string again.
- 5 Turns off the DTR signal to hang up the telephone, and assumes that result codes will be returned at whatever the default baud rate is set to. Each time DTR is turned off, BBS-PC! sends the modem the setup string again.
- 6 Transmits "true break" to hang up the telephone, and assumes that result codes will be returned at the same baud rate as the last connection. Each time true break is sent, BBS-PC! sends the modem the setup string again.
- 7 Transmits "true break" to hang up the telephone, and assumes that result codes will be returned at the default baud rate. Each time true break is sent, BBS-PC! sends the modem the setup string again.
- 8 Turns off the DTR signal to hang up the telephone, and assumes that result codes will be returned at the same baud rate as the last connection. Ignores the status of pin 6.
- 9 Turns off the DTR signal to hang up the telephone, and assumes that result codes will be returned at whatever the default baud rate is set to. Ignores the status of pin 6.
- 10 Transmits "true break" to hang up the telephone, and assumes that result codes will be returned at the same baud rate as the last connection. Ignores the status of pin 6.
- 11 Transmits "true break" to hang up the telephone, and assumes that result codes will be returned at the default baud rate. Ignores the status of pin 6.
- 12 Turns off the DTR signal to hang up the telephone, and assumes that result codes will be returned at the same baud rate as the last connection. Each time DTR is turned off, BBS-PC! sends the modem the setup string again. Ignores the status of pin 6.
- 13 Turns off the DTR signal to hang up the telephone, and assumes that result codes will be returned at whatever the default baud rate is set to. Each time DTR is turned off, BBS-PC! sends the modem the setup string again. Ignores the status of pin 6.
- 14 Transmits "true break" to hang up the telephone, and assumes that result codes will be returned at the same baud rate as the last connection. Each time true break is sent, BBS-PC! sends the modem the setup string again. Ignores the status of pin 6.

- 15 Transmits "true break" to hang up the telephone, and assumes that result codes will be returned at the default baud rate. Each time true break is sent, BBS-PC! sends the modem the setup string again. Ignores the status of pin 6.

If you are using a Hayes Smartmodem 300 or a Smartmodem 1200, set modem type to "0". For a Hayes Smartmodem 2400, set this to "1". For a list of additional modems, read further in the appendix.

The setup string can be any string of up to 40 ASCII characters. This can be used to configure the modem and set any parameters not controlled via DIP switches. Within this string, you have the following control characters available to you:

#### Char Function

- | Vertical bar. Sends a carriage return to the modem.
- \ Backslash. Sends a line feed to the modem.
- ^ Carat. Sends the next character's "control" value. For example, ^A would send CTL-A.
- Tilde. Causes a one second delay before sending the next character in the string.
- ` Reverse apostrophe. Causes a 250ms delay between each character following in the string. (Slows it down to typing speed.)

NOTE: The backslash and vertical bar are on the same key, as are the tilde and the reverse apostrophe. The former is on the lower left hand side of the keyboard and the latter is on the middle right. The carat is located above the "6" key.

What you enter for your setup string depends on how you decide you are going to detect baud rate. If you are going to use result codes, then the modem must be in the "verbose" mode. If you are going to use the hardware detection with pin 6 (DSR), you want the modem in the "non-verbose" mode.

IMPORTANT NOTE: If you are using the hardware method of detecting baud rate, do NOT program in any result codes.

We will divide these setup strings by modem and style (verbose vs. non-verbose). Decide which you need based on whether you are going to use result codes or hardware baud rate detect.

A note on detecting baud rate with pin 6. Most modems that give you a signal to indicate 1200 bps use pin 12 of the RS-232 interface to do that. The serial port of the IBM PC cannot "read" the status of pin 12. Therefore, we must move this signal to another pin on the RS-232 interface. Since using the hardware detection requires that either a special adapter be constructed to move pin 12 of the modem to pin 6 of the computer or that you modify a cable to do so, you may want to use the result codes if you have not already modified your cable. Result code detection of baud rate will work with a completely normal cable.

The setup strings are:

Hayes Smartmodem 300 verbose mode (with result codes)  
ATS2=255QO|

Hayes Smartmodem 1200 verbose mode (with result codes)  
ATS2=255QOXI|

Hayes Smartmodem 1200 non-verbose mode (with modified cable)  
ATS2=255|

Hayes Smartmodem 2400 verbose mode (with result codes)  
ATS2=255QOXI|

NOTE: The Hayes Smartmodem 2400 cannot indicate all three supported baud rates (300, 1200, and 2400) without using the result codes. Pin 12 from the modem merely reflects high speed in the current MODE (Bell vs. CCITT).

Each of these setup strings must be terminated with a vertical bar (|) character. This tells BBS-PC! to send a carriage return at that point. On the IBM PC keyboard, the vertical bar character is located above the backslash (\) and accessed with the SHIFT key.

Once default baud, modem type, setup string, and result codes (if you are going to use them) are all configured, your modems should be ready to go. If you are using a command driven modem and you will be implementing a terminal dial-out mode, you'll need to configure the dial prefix and suffix as well. The same control codes that function in the modem setup string will function in these locations.

Now that we have the general installation instructions out of the way, let's move on to a summary of modems and more modems.

#### Anchor Automation Signalman MK XII (aka Volksmodem 12)

In our testing, this modem did NOT use either the true break signal or the DTR line to hang up the telephone. Positive control of the status of the phone line is essential to BBS operation and cannot be assured without some sort of logic signal control (command strings are too unreliable, since they depend on whether or not data is being received, etc.).

We feel, therefore, that the Signalman Mark XII modem is not well-suited for use in an host environment (although it may function FINE as a terminal), and cannot recommend it for use with BBS-PC!.



Radio Shack's DC-2212

Quite a few of our Model 1000, 1200HD, and 2000 owners may be operating with the DC-2212. BBS-PC! can be configured to work with this modem.

This modem does not report the baud rate of the incoming phone call when it is in its factory configuration. To correct this, you should have the "RS-232 Extended Protocol" modification made by a Radio Shack technician. This modification is described on page 15 of the modem service manual and involves closing jumper W-1.

When this is complete, the DC-2212 will (like the Hayes) report baud rate on pin 12 of the RS-232 interface. You will need to modify your cable or build an adapter box to move this signal to pin 6. (DSR)

Default baud should be "300" and modem type should be "0". There are no result codes with this modem, since it is either hardware detection or no detection at all.

The modem settings for a DC-2212, then, would appear as:

```
Modem #0
A: Default baud: 300
B: Modem type : 0
C: Setup string: `*.C~*.GA
D: Dial prefix : `*.DT
E: Dial suffix : `X
F: 300 baud :
G: 1200 baud :
H: 2400 baud :
Modem #1
I: Default baud: 0
J: Modem type : 0
K: Setup string:
L: Dial prefix :
M: Dial suffix :
N: 300 baud :
O: 1200 baud :
P: 2400 baud :
```

If you happen to have a DC-2212 that does NOT have the modification for 1200 bps call reporting, you can use a modem type of "8" instead of "0". This tells BBS-PC! to ignore the status of the pin 6 and expect every call to be the same as the default baud. This lets you answer 300 or 1200 baud calls, but not both.

Radio Shack's Model 1000 internal modem board

This modem is 300 bps only and is roughly compatible with the DC-2212. The only difference will be in the setup strings. Settings for this modem appear as:

```
Modem #0
A: Default baud: 300
B: Modem type : 10
C: Setup string: `*C`*GMAX
D: Dial prefix : `*DT
E: Dial suffix : `X
F: 300 baud :
G: 1200 baud :
H: 2400 baud :
Modem #1
I: Default baud: 0
J: Modem type : 0
K: Setup string:
L: Dial prefix :
M: Dial suffix :
N: 300 baud :
O: 1200 baud :
P: 2400 baud :
```

Please note that the modem type is also different from the DC-2212.

Ven-Tel half card PC internal modem

For the most part, this modem is Hayes compatible. The only difference is that you must send the modem the setup string each time you terminate a phone connection. For that reason, set your modem type to "4".

In addition, this modem lacks the 8 DIP switches of the Hayes. Therefore, you will use the setup strings to control ALL areas of modem configuration. The settings are:

```
Modem #0
A: Default baud: 1200
B: Modem type : 4
C: Setup string: ATSO=1S2=255EQOV1X1|
D: Dial prefix : ATDT
E: Dial suffix : |
F: 300 baud : CONNECT|\
G: 1200 baud : CONNECT 1200|\
H: 2400 baud : CONNECT 2400|\
Modem #1
I: Default baud: 0
J: Modem type : 0
K: Setup string:
L: Dial prefix :
M: Dial suffix :
N: 300 baud :
O: 1200 baud :
P: 2400 baud :
```

If your modem is set for COM2 addressing, you should use the Modem #1 position instead. Also, even though the Ven-Tel will never establish a connection at 2400 bps, the result code position needs to be filled. If one result code is present, they ALL must be.

#### AST Reach modem

In our testing, this modem sends the result codes to the computer at the bit rate of the INCOMING phone call. Since there is no way for the BBS to know what this bit rate is, it cannot correctly read the result codes.

We feel, therefore, that the AST Reach modem is not well-suited for use in an host environment (although it may function FINE as a terminal), and cannot recommend it for use with BBS-PC!.

#### Radio Shack Modem II

There are three rocker switches on the front panel on the Model II. The "Man/Auto" switch should be set to the "Auto" position, the "Org/Ans" switch should be set to the "Ans" position, and the "Test" switch should be set to the "Off" position. Make VERY sure that the "Force DTR" switch located on the back of the modem is set to the "Off" position.

When you power up the Modem II, the CD light should NOT be lit. If it is, power the modem off and back on again. Remember that the Modem II has a 6 second delay period before the modem stabilizes. Give it this delay before you boot up the BBS program.

There are no result codes or setup strings for the Modem II. It is 300 bps only, so there is nothing to detect. The "Auto" switch makes the modem answer the phone, so there is nothing to set up.

The settings are:

```
Modem #0
A: Default baud: 300
B: Modem type : 10
C: Setup string:
D: Dial prefix : `*DT
E: Dial suffix : `X
F: 300 baud :
G: 1200 baud :
H: 2400 baud :
Modem #1
I: Default baud: 0
J: Modem type : 0
K: Setup string:
L: Dial prefix :
M: Dial suffix :
N: 300 baud :
O: 1200 baud :
P: 2400 baud :
```

#### Prometheus Promodem 1200

As with the Ven-Tel, this modem seems fairly Hayes compatible. Experience with our customer base would seem to indicate that the modem should be set for single digit result codes instead on English words. It seems to return those more reliably.

The modem settings are:

```
Modem #0
A: Default baud: 1200
B: Modem type : 4
C: Setup string: ATSO=1S2=255EQOV0X1|
D: Dial prefix : ATDT
E: Dial suffix : |
F: 300 baud : 1
G: 1200 baud : 5
H: 2400 baud : NOT USED
Modem #1
I: Default baud: 0
J: Modem type : 0
K: Setup string:
L: Dial prefix :
M: Dial suffix :
N: 300 baud :
O: 1200 baud :
P: 2400 baud :
```

Note that as with the Ven-Tel, the 2400 bps slot is filled with a dummy string. Make certain that you set your Promodem up to "honor DTR". They don't come from the factory this way, but it is controlled by a DIP switch.

We have tested the Prometheus Promodem 1200 external in our office. We cannot get the modem to operate in the same manner twice. Sometimes it responds to DTR and sometimes it does not, with NO changes to either the software configuration or the modem DIP switches. Sometimes it would return the extended result code set and sometimes it would not. And there was NEVER any rhyme or reason to it.

We, therefore, do not recommend this modem for reliable BBS-PC! operation. While it DID work some of the time, we were unable to pinpoint what conditions affected this.

### Hayes and 100% compatibles

We list the following modems as tested with BBS-PC! and they seem to be completely Hayes compatible:

US Robotics Courier 2400/Microlink 2400  
MultiTech Systems MultiModem224  
Racal Vadic PC Modem  
Hayes Smartmodems (of course)

All of these modems worked well with BBS-PC! and are on the top of the list of recommended modems.

Earlier in this section, we addressed the switch settings and setup strings for the Hayes Smartmodems. The US Robotics and MultiTech modems will be the same modem type, setup string, and switch settings as the Hayes 1200 (even though they are 2400 baud modems). The US Robotics and MultiTech modems each have additional DIP switches past the 8 that we list for Hayes. Leave these set in their factory configuration. Set the initial 8 as compatible with the Hayes 1200 and ignore the rest.

The modem settings for the Hayes 2400 should be:

```
Modem #0
A: Default baud: 2400
B: Modem type : 1
C: Setup string: ATS2=255QOX1|
D: Dial prefix : ATDT
E: Dial suffix : |
F: 300 baud : CONNECT|\
G: 1200 baud : CONNECT 1200|\
H: 2400 baud : CONNECT 2400|\
Modem #1
I: Default baud: 0
J: Modem type : 0
K: Setup string:
L: Dial prefix :
M: Dial suffix :
N: 300 baud :
O: 1200 baud :
P: 2400 baud :
```

Please note that this assumes that you have already set up the permanent RAM of the Hayes 2400 as described at the beginning of this appendix. If you have not, do so now.

The US Robotics, MultiTech, and Hayes 1200 settings are:

```
Modem #0
A: Default baud: 1200
B: Modem type : 0
C: Setup string: ATS2=255QOX1|
D: Dial prefix : ATDT
E: Dial suffix : |
F: 300 baud : CONNECT|\
G: 1200 baud : CONNECT 1200|\
H: 2400 baud : CONNECT 2400|\
Modem #1
I: Default baud: 0
J: Modem type : 0
K: Setup string:
L: Dial prefix :
M: Dial suffix :
N: 300 baud :
O: 1200 baud :
P: 2400 baud :
```

The US Robotics and MultiTech modems can have the default baud increased to "2400" (optionally). For a Hayes 300, refer to the start of this appendix. If you are using a Racal Vadic PC modem, use the above settings with a modem type of "1". Set the switches per the Hayes instructions.

Note to those using pin 6 sensing

If you are using the pin 6 method of detecting 1200 bps, you should remove ALL result codes from your modem defaults. These are not needed and will cause incorrect operation.

If you are using a Rixon modem in the Hayes compatible mode, make note of the fact that your modem does NOT support the extended result codes to indicate bit rate of incoming calls. You will need to use the pin 12 to pin 6 method of bit rate detection.





## **Micro-Systems Software's BBS-PC Features List**

- True message re-formatting to callers terminal. Messages are always displayed to callers terminal width, regardless of the original width of the text.
- Private messages. Users may send private mail that may be read only by the addressee.
- Multi-level system security. BBS-PC allows up to 256 levels of security in both the message base and the upload/download database. Each user may be "enabled" or "disabled" for read, write, upload, download access in up to 16 sections.
- Permanent user records. BBS-PC "remembers" each regular user and stores terminal information, security access levels, last message read and other information.
- Individual user passwords. BBS-PC users may assign themselves a password which prevents system access by unauthorized callers.
- File uploading/downloading. Users may upload or download files to BBS-PC in any of 4 protocols, including XMODEM, and standard ASCII in both line and/or block transmission modes.
- BBS-PC is almost 100% self-maintaining. Message space is automatically reclaimed when messages are deleted and messages will "roll" if necessary to support new entries. SYSOP may control the maximum number of messages, file uploads, and user in the user log.
- Dual mode operation. BBS-PC is capable of operating in both "open" and "closed" modes. Open mode allows users to add themselves as members, while closed mode offers a higher level of security.
- Screen saver. While awaiting a caller BBS-PC will automatically disable the display screen after 5 minutes of inactivity, minimizing screen burn.
- DOS directory isolation. Users are 100% isolated from the host directory and DOS environment. Upload/Download files are maintained by the system's own higher-level software directory.
- Hacker-proof. BBS-PC was designed from the outset to be as hacker-proof as possible. Thus far, the system has NEVER been "crashed".
- Instant carrier loss recovery. BBS-PC does not require a host "reset" on carrier loss or other such nonsense. Complete recovery is easily accomplished by the system, even during file uploads and downloads.
- Support for two modems. BBS-PC supports 1 or 2 command driven modems on either COM port. With optional "dial-out" feature for relaying calls.
- Time clock protection. System supports both call limiting and a "sleep" timeout for inactivity. These times are configurable.
- Completely configurable. Options to configure almost all aspects of the system. Including COM parameters, linefeed options, message base size, member base size, etc. Also allows separate directory paths for these files.
- Message and file upload queueing. Allows SYSOP policing of both messages and uploads BEFORE general access is permitted.
- FAST. By using complex ISAM routines, BBS-PC is one of the fastest systems available. Easily maintains 1200 or 2400 baud.

# Micro-Systems Software's BBS-PC Features List

- True message re-formatting to callers terminal. Messages are always displayed to callers terminal width, regardless of the original width of the text.
- Private messages. Users may send private mail that may be read only by the addressee.
- Multi-level system security. BBS-PC allows up to 256 levels of security in both the message base and the upload/download database. Each user may be "enabled" or "disabled" for read, write, upload, download access in up to 16 sections.
- Permanent user records. BBS-PC "remembers" each regular user and stores terminal information, security access levels, last message read and other information.
- Individual user passwords. BBS-PC users may assign themselves a password which prevents system access by unauthorized callers.
- File uploading/downloading. Users may upload or download files to BBS-PC in any of 4 protocols, including XMODEM, and standard ASCII in both line and/or block transmission modes.
- BBS-PC is almost 100% self-maintaining. Message space is automatically reclaimed when messages are deleted and messages will "roll" if necessary to support new entries. SYSOP may control the maximum number of messages, file uploads, and user in the user log.
- Dual mode operation. BBS-PC is capable of operating in both "open" and "closed" modes. Open mode allows users to add themselves as members, while closed mode offers a higher level of security.
- Screen saver. While awaiting a caller BBS-PC will automatically disable the display screen after 5 minutes of inactivity, minimizing screen burn.
- DOS directory isolation. Users are 100% isolated from the host directory and DOS environment. Upload/Download files are maintained by the system's own higher-level software directory.
- Hacker-proof. BBS-PC was designed from the outset to be as hacker-proof as possible. Thus far, the system has NEVER been "crashed".
- Instant carrier loss recovery. BBS-PC does not require a host "reset" on carrier loss or other such nonsense. Complete recovery is easily accomplished by the system, even during file uploads and downloads.
- Support for two modems. BBS-PC supports 1 or 2 command driven modems on either COM port. With optional "dial-out" feature for relaying calls.
- Time clock protection. System supports both call limiting and a "sleep" timeout for inactivity. These times are configurable.
- Completely configurable. Options to configure almost all aspects of the system. Including COM parameters, linefeed options, message base size, member base size, etc. Also allows separate directory paths for these files.
- Message and file upload queueing. Allows SYSOP policing of both messages and uploads BEFORE general access is permitted.
- FAST. By using complex ISAM routines, BBS-PC is one of the fastest systems available. Easily maintains 1200 or 2400 baud.

